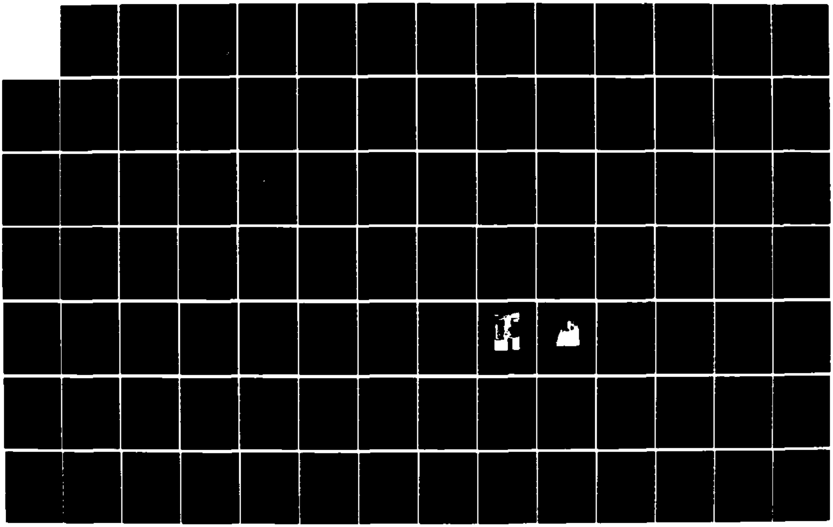
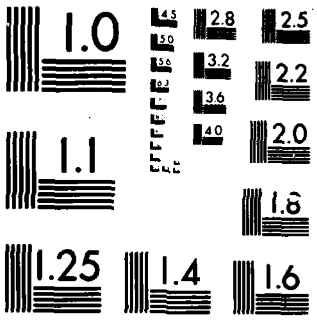
AD-A164 018 PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD 1/2
CRACK GROWTH IN A H..(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. R L HASTIE
UNCLASSIFIED DEC 85 AFIT/GA/AA/85D-6 F/G 11/6 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD CRACK GROWTH IN A HIGH-TEMPERATURE SUPERALLOY

THESIS

Robert L. Hastie Jr.
Captain, USAF

AFIT/GA/AA/85D-6

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GA/AA/85D-6

PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD CRACK GROWTH IN A HIGH-TEMPERATURE SUPERALLOY

THESIS

Robert L. Hastie Jr.
Captain, USAF

AFIT/GA/AA/85D-6

Approved for public release; distribution unlimited

AFIT/GA/AA/85D-6

PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD CRACK GROWTH IN A HIGH-TEMPERATURE SUPERALLOY

THESIS

Presented to the Faculty of the School of Engoneering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Master of Science in Astronautical Engineering

Robert L. Hastie Jr., B.S.
Captain, USAF

December 1985

Approved for public release; distribution unlimited

Preface

The purpose of this study was to support the Engine Structural Integrity Program and the Retirement-for-Cause maintenance program. Both programs have been initiated by the USAF to extend the useful service life of future and present engine components. Successful implementation of both programs requires accurate analytical methods for predicting crack growth. I personally found it rewarding to develop a analytical technique for predicting sustained-load crack growth after overloads in engine components.

This study, however, would not have been possible without all the help and support I received. I wish to express my sincere thanks and gratitude to Dr. T. Nicholas AFWAL/MLLN and his department for the use of their facilities and their time and efforts in assisting me. In particular I would like to thank Mr. G. Ahrens and Mr. W. Goddard, UDRI, for their help setting up the experimental test apparatus. I also thank Mr.G. Hartman and Mr. D. Johnson for helping me overcome the difficulties learning a new computer system. In addition, I appreciated Capt K. Harms explaining his previous work to me. I also greatly appreciated the overall guidance and support my advisor Major G. K. Haritos, AFIT/ENY, provided during this study.

I especially thank my wife Victoria who provided immeasurable support and encouragement when I needed it.

A-1

Table of Contents

List of Figures

Figure Page

List of Tables

Abstract

This study investigates methods of modeling the effects of overloads on high-temperature sustained-load crack growth. In addition to a model previously developed for this specific problem, a computer program developed for low-temperature, high-frequency cyclic load applications was evaluated. Sustained-load hold times were converted to equivalent fatigue cycles to analyze a load spectrum, consisting of sustained-load with periodic overloads. The CRACKS crack growth program was used with the Wheeler and Willenborg models used to account for crack growth retardation due to overloads.

Predictions were compared with experimental test data generated on specimens of Inconel 718 at 650 C with periodic overloads of either 20 or 50 percent. Crack measurements were made using a electric potential system. The application of the electric potential system to crack growth measurement following overloads was extensively evaluated. It was concluded that the system had to be recalibrated after each overload due to a sudden advancement in crack length.

The retardation models were found to require empirical parameters that depend upon the stress intensity level for

each overload application. Using relationships developed for these parameters, the CRACKS program using the Wheeler model was found to be capable of predicting the time-to-failure for sustained-loading with periodic overloads within 20 percent of test data. The Willenborg model was found to be inapplicable to this problem because it depends solely on stress ratio which has no physical meaning for sustained-loading. The Wheeler model, on the other hand, could generally be applied to sustained-load crack growth using equivalent fatigue cycles. In conjunction with the CRACKS computer program, this could provide a powerful new method for evaluating crack growth under general engine mission spectra including the effects of overloads.

PREDICTING THE EFFECTS OF OVERLOADS
ON SUSTAINED-LOAD CRACK GROWTH
IN A HIGH-TEMPERATURE SUPERALLOY

I. Introduction

Background

Design trends for modern engines have emphasized increased performance with higher thrust/weight ratios, while also requiring improved engine durability and maintainability. Engine components must therefore be designed to endure the increasing severity in operating conditions. Until recently, the method used to predict the useful service life of engine components was based on statistical life models. This method was very conservative and would retire an entire population of engine disks when it was predicted that, statistically, 1 in 1000 disks would develop a 0.03-inch fatigue-induced crack [1]. Although this method helped preclude disk failures, significant useful life remained in the other 999 disks retired. Estimates placed this residual useful life at greater than 10 more lifetimes for 80 percent of the disks retired [2].

Under the new "Retirement-for-Cause" concept initiated by the USAF, this additional useful life can be utilized by adopting an inspection criterion applied to components after a specific period of time. If the inspected components pass this criterion, they may be returned to service. The criterion is based on fracture mechanics calculations to determine what minimum crack size, if undetected, would grow to failure before the next inspection.

Fracture mechanics is also the basis for predicting crack growth as required by the Engine Structural Integrity Program (ENSIP) specification [3]. The ENSIP specification requires a damage tolerant design approach be applied to structural critical components on all future USAF engine designs. Under this approach, initial flaws or defects are assumed to exist in the components when they enter service. Analysis of components along with verification testing must demonstrate that the initial flaws will not grow to a catastrophic size within the design lifetime of the component.

It is clear that successful implementation of the Retirement-for-Cause program and the ENSIP design approach depends upon technical capability in two key areas. The first is a Nondestructive Evaluation (NDE) procedure used to determine the largest initial flaw size existing in a component after an inspection. This initial flaw size is then assumed to exist in all components. The second is the capability to use analytical prediction models to accurately predict the crack propagation from the initial crack size to failure.

There are numerous crack growth rate prediction models with varying degrees of complexity. Most of these models have been developed for application to airframe components under typical airframe spectra including large numbers of cycles with periodic overloads. For engine applications, the spectra are simpler, involving fewer cycles and only occasional overloads. No complex interaction models have been developed for engine spectrum loading which involves both cyclic and sustained-loading.

Simple crack growth models usually calculate the growth cycle-by-cycle by integrating a crack growth rate equation. More complex models used in airframe analysis include retardation routines to account for the decrease in growth rate following a peak overload cycle. These models do not address crack growth under sustained (hold-time) loading which is present in a typical engine spectrum. CRACKS [4] which represents the state-of-the-art in airframe spectrum crack growth analysis is a complex prediction program used to analytically calculate crack growth under large spectrums of cyclic loading. This program includes the Wheeler and Willenborg models for predicting retardation effects.

Objective

This thesis explores one aspect of the complex crack-growth-rate prediction problem. Specifically, this thesis explores the applicability of existing crack-growth retardation models, developed for high-frequency, low-temperature airframe applications, to high-temperature sustained-load crack growth retardation. Procedures will be developed to convert sustained-load time to equivalent fatigue cycles so that classical Wheeler and Willenborg retardation models can be used in the CRACKS computer program to predict sustained-load crack growth rates following overloads. The retardation models will be applied to data obtained from previous experimental work as well as new experimental proof tests to verify each model's capability.

II. Retardation Model Theory

In this study three plastic zone retardation models were examined. First was the Wheeler model [5] which reduces the crack growth rate da/dn to account for retardation. Second was the Willenborg model [6] which accounts for retardation by reducing the maximum and minimum stress intensity factors. Also, the minimum stress intensity factor, if negative, is truncated at zero. Finally was the Overload model developed by K. Harms, T. Weerasooriya, and T. Nicholas [7] [8] which accounts for retardation by reducing the stress intensity factor K, to a lower effective value K_{eff}, after each overload. The theory behind each of these models is discussed in the following sections.

Wheeler Model

The basis for the Wheeler, Willenborg, and Overload models is that an overload cycle produces an extended plastic zone that retards crack growth. Thus, in figure 1, r_p represents the plastic zone due to sustained-loading and $\overline{r}_p$ represents the plastic zone due to an applied overload. As long as a_1 is less than a_2, the sustained-load crack is growing in the overload plastic zone at a slower or retarded rate. The Wheeler model predicts this retardation effect by reducing the crack growth rate while growing through the overload plastic zone. The crack growth rate da/dn is reduced by multiplication with a retardation parameter C_p. This parameter depends on the ratio of the current sustained-load plastic zone size to the previous overload plastic zone size raised to a shaping exponent m. If the sustained-load plastic zone grows past the prior overload plastic zone, no retardation is predicted. In terms of the symbols used in figure 1, the retardation parameter is defined as,

$$C_p = \left(\frac{r_p}{a_2 - a_1 + r_p} \right)^m \qquad \text{for } a_1 < a_2 \qquad (1)$$

and

$$C_p = 1 \qquad \text{for } a_1 \geq a_2 \qquad (2)$$

where r_p = extent of current yield zone

$a_2 - a_1 + r_p$ = distance from crack tip to elastic plastic interface

m = shaping exponent

The shaping exponent m is used to calibrate the retardation model with experimental data. Generally, m has been found to be a material dependent constant.

Once the shaping exponent is defined for the given material, the value of C_p is substituted into equation 3.

$$\frac{da}{dn}\text{(retarded)} = C_p \frac{da}{dn}\text{(non-retarded)} \qquad (3)$$

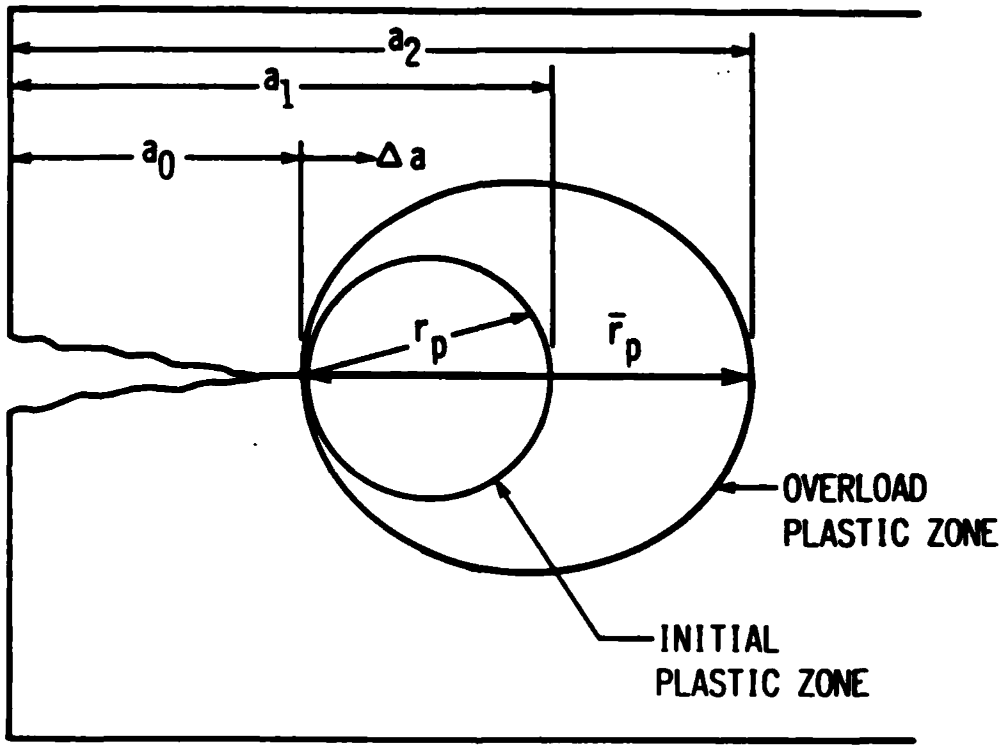


Figure 1 Schematic diagram of the plastic zones at the crack tip.

The Wheeler model accounts for retardation by substituting da/dn (retarded) for da/dn (non-retarded) while the crack is growing through the overload plastic zone. The numerical procedures used to implement the Wheeler model are contained in the CRACKS program.

Willenborg Model

The Willenborg model like the Wheeler model is based on yield zone analyses. The retardation effect is accounted for by a reduction in stress intensity factor. The reduction factor is calculated by first finding an equivalent stress, σ_{ap}, that will produce a plastic zone from the current crack tip location to the edge of the overload plastic zone. Refering to figure 1, this corresponds to a plastic zone of radius of $\overline{r}_p - \Delta a$. Second, a stress reduction factor, σ_{red}, is calculated by subtracting the currently applied maximum stress σ_{max} from σ_{ap}.

$$\sigma_{red} = \sigma_{ap} - \sigma_{max} \tag{4}$$

When the crack growth amount Δa plus the current plastic zone size r_y equals the overload plastic zone size $\overline{r}_y$, the value of σ_{red} is set equal to zero, since the crack propagation is no longer retarded. Third, effective values of the currently applied stresses are calculated by:

$$[\sigma_{max}]_{eff} = \sigma_{max} - \sigma_{red} \qquad (5)$$

$$[\sigma_{min}]_{eff} = \sigma_{min} - \sigma_{red} \qquad (6)$$

If either effective stress is less than zero, it is set equal to zero. Finally, using the effective maximum and minimum stress values, effective stress intensity factors are calculated. This produces the following retardation relationships.

$$[K_{max}, K_{min}]_{eff} = [K_{max} - K_{red} \, , \, K_{min} - K_{red}] \qquad (7)$$

$$\text{where } K_{red} = K_{ap} - K_{max} \qquad \text{for } a_1 < a_2$$

$$K_{red} = 0 \qquad \text{No retardation for } a_1 \geq a_2$$

After each crack growth increment a new value of σ_{ap} is calculated. The corresponding new value of K_{red}, calculated using equation (4), is substituted into equation (7), yielding the new K_{eff} values for the next growth increment.

Overload Model

This model uses a linear cumulative damage concept to sum the growth contributions of a single overload fatigue cycle and growth due to sustained load. The basis for the Overload model, like the Wheeler and Willenborg models, is that an overload cycle produces an extended plastic zone that retards crack growth. The plastic zone concept is illustrated schematically in figure 1. The sustained load and overload fatigue cycle have stress intensity factors

K_s and K_m and plastic zone radii denoted by r_p and $\overline{r}_p$ respectively. The distances from the center of the crack to the edge of the plastic zones due to sustained loading and a fatigue overload cycle applied when the crack length was a_o are labeled a_1 and a_2. For a crack advancement Δa from a_o, the crack tip will be in an overload plastic zone until $\Delta a = a_2 - a_1$. While in this plastic zone the growth rate will be retarded. The Overload model uses a reduced value of stress intensity factor, K_{eff}, to account for the retarded growth rate. For modeling purposes, K_{eff} is taken in the form:

$$K_{eff} = K_s \left[1 - \alpha \exp(-\beta \Delta a) \right] \tag{8}$$

where K_{eff} = *effective (reduced) stress intensity factor*

K_s = sustained-load stress intensity factor

α, β = modeling parameters

Δa = incremental crack extension

The parameter β is chosen such that steady-state crack growth will resume after the crack has traversed the overload plastic zone. Mathematically, it is desired to have K_{eff} approach K_s when Δa approaches $(\overline{r}_p - r_p)$. This is accomplished by letting

$$\beta \Delta a = \pi \sqrt{2}. \tag{9}$$

When $\Delta a = \overline{r}_p - r_p$, the resulting value of K_{eff}/K approaches unity to within one percent as shown in figure 2. The plane stress plastic zone sizes for the overload cycle and sustained load are given by,

$$\begin{aligned} \overline{r}_p &= [\, K_m / \sigma_y \,]^2 / \pi \\ r_p &= [\, K_s / \sigma_y \,]^2 / \pi \end{aligned} \qquad (10)$$

where σ_y is the uniaxial tension yield stress of the material. Substituting $\Delta a = \overline{r}_p - r_p$ and equation (10) into equation (9) yields an expression for β:

$$\beta = \frac{\sqrt{2}\ \pi^2\ \sigma_y^{\,2}}{K_s^{\,2}\ (\ \tau^{\,2} - 1)} \qquad (11)$$

where the overload ratio τ is defined by:

$$\tau = \frac{K_m}{K_s} \qquad (12)$$

Observing equation (11) it is noted β is only a function of material properties and test conditions. β therefore cannot be used as an adjustable parameter to fit experimental data. This leaves the parameter α given in equation (8), to be adjusted to fit experimental data. The value of α chosen determines K_{eff} immediately after an overload application and therefore can be used to model the reduced value of sustained-load crack growth rate. The effect of varying α on the ratio of K_{eff}/K_s is seen in figure 2. Also this figure demonstrates how the Overload

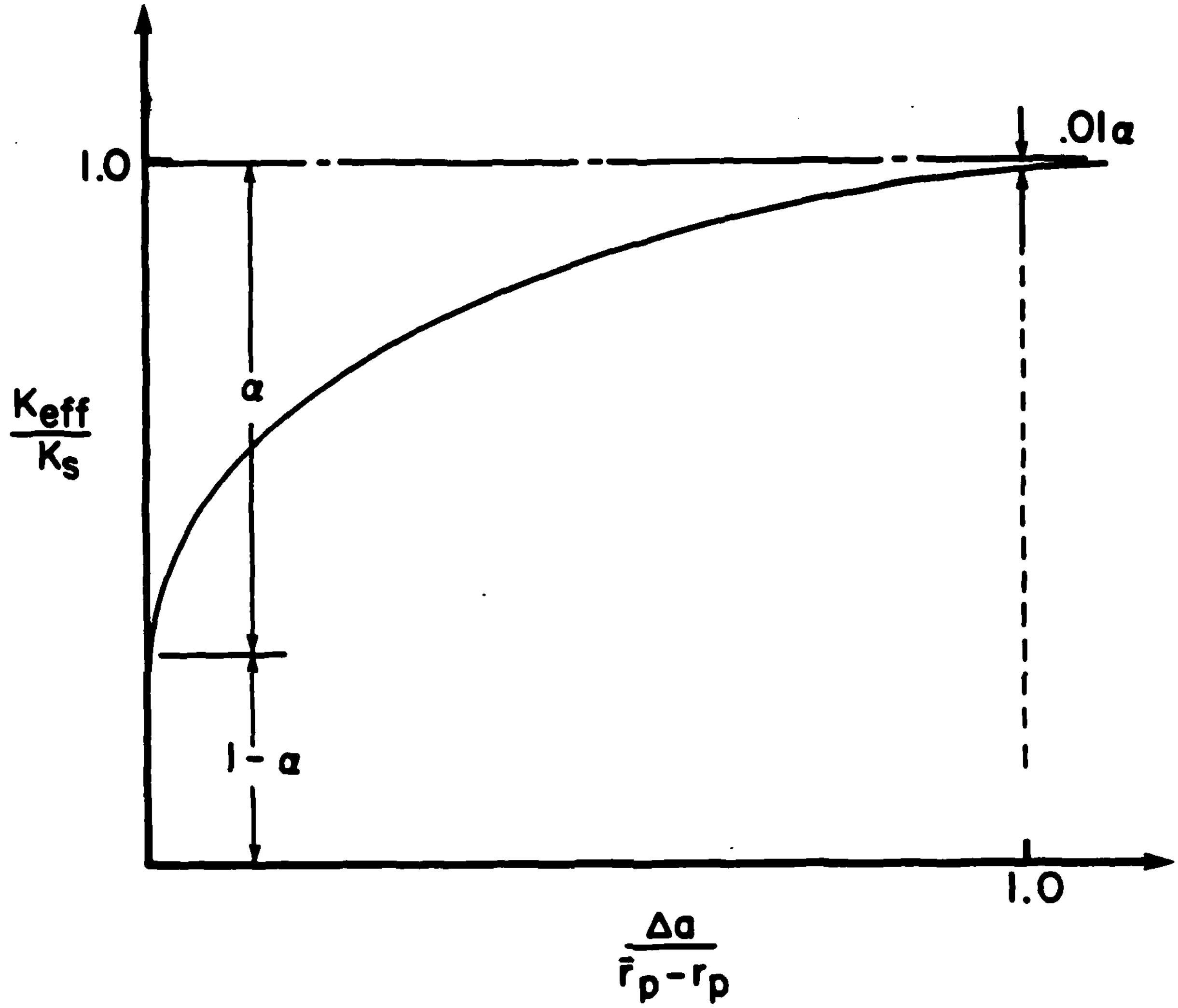


Figure 2 Variation of K_{eff}/K_s as a function of normalized crack advancement from the crack tip in the overload plastic zone.

model uses a decreasing exponential to asymptotically approach the normal crack growth.

In a prior investigation, Harms made experimental measurements of the delay time before normal sustained-load crack growth resumed after an overload was applied. Harms [7] noted that delay times and, hence, the value of K_{eff} changed systematically with K. Plots of the delay times Δt_r versus K level for the 20% and 50% overload cases tested by Harms are shown in figures 3 and 4. The boundaries of the data were used to approximate the average delay time (Δt_{avg}) curve. Using an iteration process, values of α were chosen with a specified value of K to generate a Δt_r for the model. Adjustments in α were made until Δt_r calculated and Δt_{avg} agreed resonably well. Harms repeated this process over the full range of K values to define a functional relationship for Δt_r, K, and α.

The functionals developed were expressed in non-dimensional terms as α/α^* and K/K^* where α^* and K^* are threshold values. K^*, a material property, was substituted into equation (1) for Keff with $\Delta a = 0$ and $\alpha = \alpha^*$. Solving for α^* yields $\alpha^* = 1 - (K^* / K_s)$ with the limiting values being:

$$\frac{\alpha}{\alpha^*} = 1 \quad \text{total crack arrest} \qquad (13)$$

$$\frac{\alpha}{\alpha^*} = 0 \quad \text{no retardation.} \qquad (14)$$

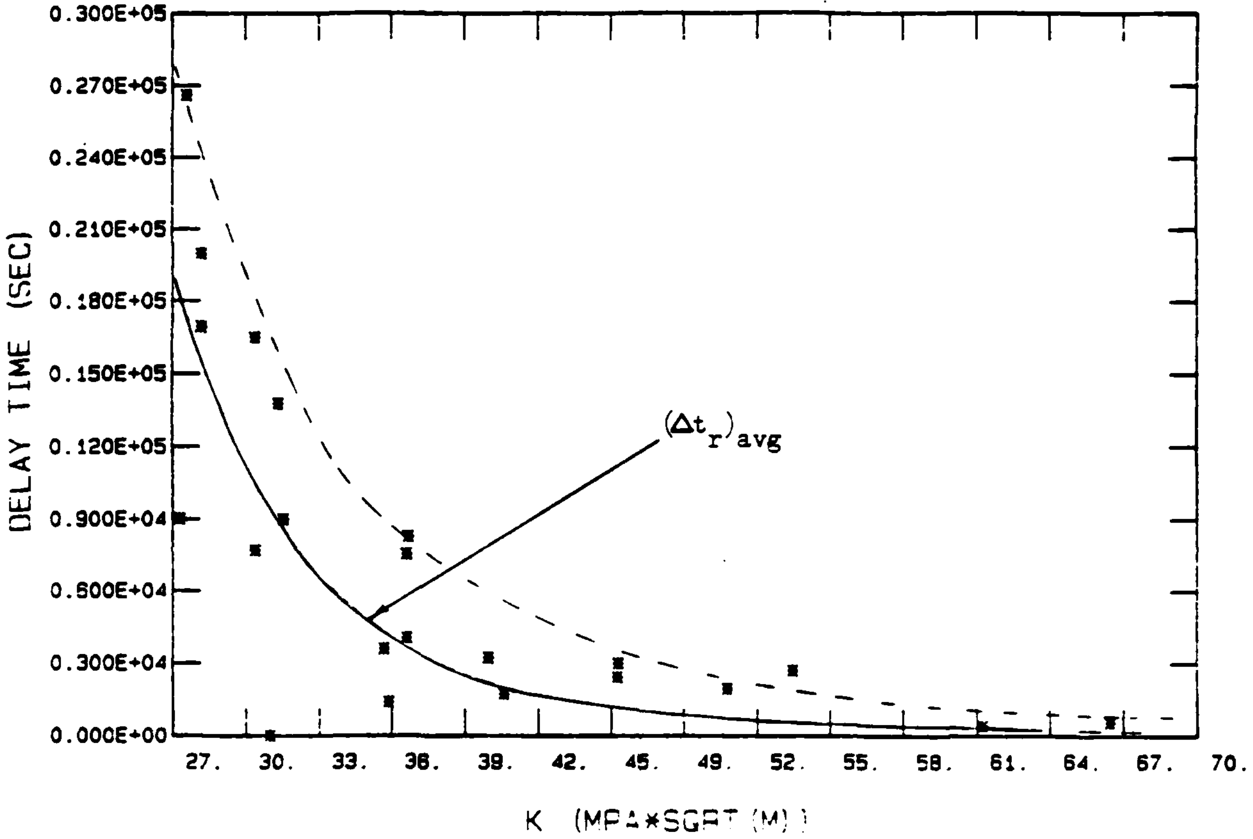


Figure 3 Delay Times Resulting from 20 Percent Overloads at Various Stress Intensities.

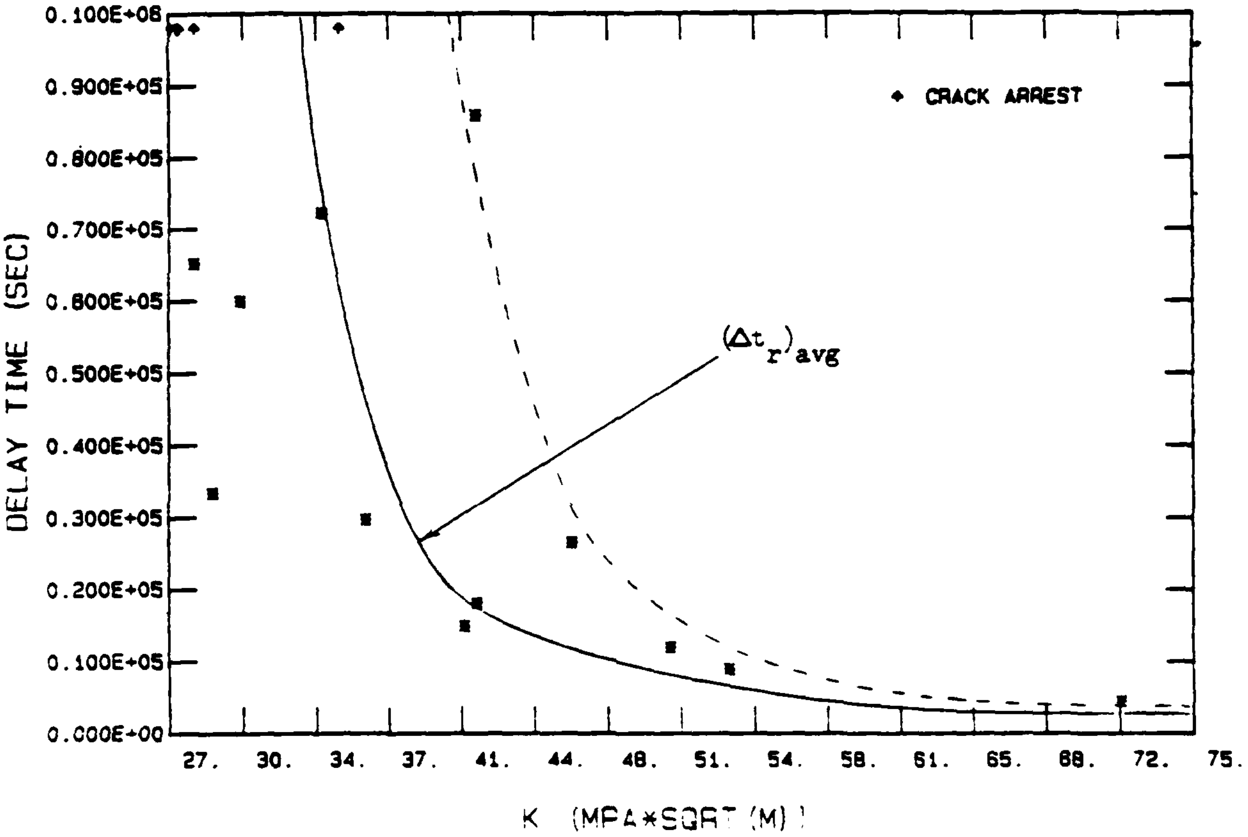


Figure 4 Delay Times Resulting from 50 Percent Overloads at Various Stress Intensities.

The α versus K curves generated by Harms for 20% and 50% overloads, normalized to the threshold values, are shown in figure 5. The curves were fit with a polynomial equation. The equation for the 20% overload case is

$$\alpha = \alpha^{*} [(-.0730791E-01) (K/K^{*})^{3} + (0.303086) (K/K^{*})^{2} -(0.422108) (K/K^{*}) + (0.117517E-01)] \qquad (15)$$

while that for the 50% overload case is

$$\alpha = \alpha^{*} [(-.121127E-01) (K/K^{*})^{2} + (0.239231E-01) (K/K^{*}) + (0.987133)]. \qquad (16)$$

With α and β defined, the Overload model's retardation effect was calculated using equation (8). The K_{eff} value obtained from this equation is used to account retardation while within the overload plastic zone.

It is apparent all three of the retardation models discussed are numerically cumbersome. Therefore, computer programs were developed to implement the retardation calculations.

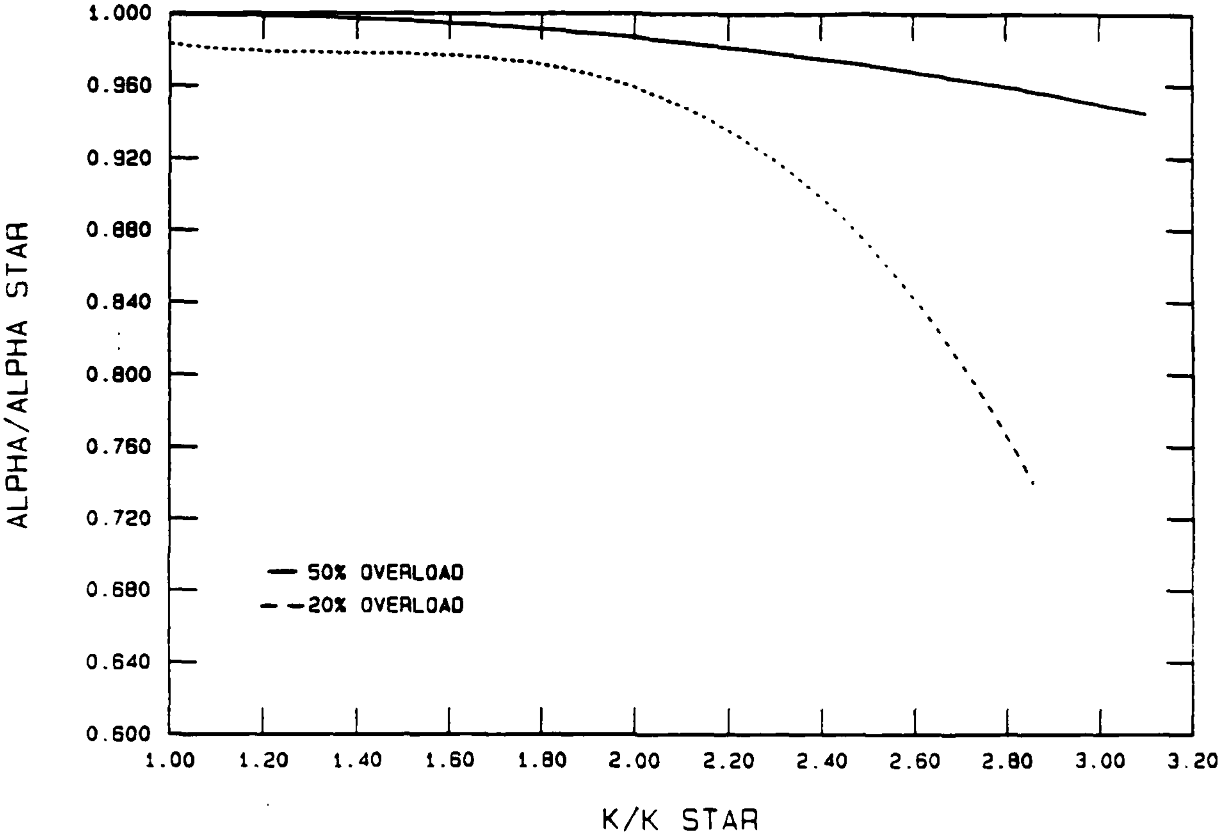


Figure 5 α Functions for 20 and 50 Percent Overloads.

III. Computer Model Development

In this study two computer programs were used to perform the numerical calculations required to predict the crack growth. First, a computer program called "Overload" was written to integrate a creep crack growth rate (da/dt) equation and incorporate the Overload retardation model. The second program used was the CRACKS program which integrates a fatigue crack growth (da/dn) equation. To use this program for sustained-load crack growth required converting sustained load time into equivalent fatigue cycles. The crack growth due to a simple spectrum of sustained-load (represented by equivalent fatigue cycles) with periodic overloads was analyzed using the CRACKS program. Both the Wheeler and Willenborg retardation schemes were used in CRACKS. After completing the growth predictions, equivalent cycles were converted back to sustained load time. The theory and assumptions associated with each model will be discussed in detail in the following sections.

Overload Program

This program was written to carry out the numerical calculations required to use the Overload retardation model. A listing of the program is contained in appendix 3. The input data for this program are the initial crack size, load history, sustained-load crack growth rate, and compact tension specimen dimensions. At anytime, for a given crack length and sustained-load amplitude, the stress intensity factor is calculated using the compact tension solution [9] below:

$$K = \frac{P \quad (2+a/w)\ f(a/w)}{b\ \sqrt{w} \quad (1-a/w)^{3/2}} \qquad (17)$$

where K = Stress Intensity factor

$f(a/w)=(0.886+4.64(a/w)-13.32(a/w)^2+14.72(a/w)^3+5.6(a/w)^4)$

a = Crack length

P = Applied load

b = Specimen thickness

w = Specimen width

Once K is known, the crack growth rate is determined from the crack growth rate equation relating da/dt to K. A Modified Sigmoidal Equation (MSE) is used to represent this relationship. The MSE model, developed by General Electric [10], uses the various coefficients in the following equation to fit a sigmoidal curve through the data.

$$da/dt = [\exp(B)]\ [K/K^*]^P\ [\ln(K/K^*)]^Q\ [\ln(K_c/K)]^D \quad (18)$$

where da/dt = Crack growth rate

K = Current stress intensity value

K^* = Threshold stress intensity value

K_c = Critical Stress intensity value

B,P,Q,D = Fitting parameters of the curve

In this equation, K^{*} and K_c are in units of (MPa $m^{1/2}$) and da/dt is calculated in units of (m/sec). The remaining constants are non-dimensional. Harms [3] determined the sigmoidal coefficients by fitting the data from his constant sustained-load baseline test. The test data, along with the best-fit MSE coefficients, are shown in figure 6.

With the relationship between a, K and da/dt known, it is possible to find the time it takes to grow from an initial crack size of a_i to final crack length a_f using:

$$\Delta t = \int_{a_i}^{a_f} \frac{da}{da/dt_{MSE}} \qquad (19)$$

This integration is carried out numerically by dividing the region of crack growth $(a_f - a_i)$ into a finite number of increments Δa and summing the time to grow each increment. For each small increment, Δa, the value of K is calculated for the end points and for any intermediate crack lengths from equation (17). Similarly, da/dt is determined for the end points and for discrete points in between using equation (18). The numerical integration of each increment Δa is performed via Simpson's rule [11]. The total time to grow from a_i to a_f is obtained by summing the time to grow each Δa interval. Since Simpson's is rule only a numerical approximation of the exact integral of the function, care was taken to ensure that proper accuracy was carried through the calculations.

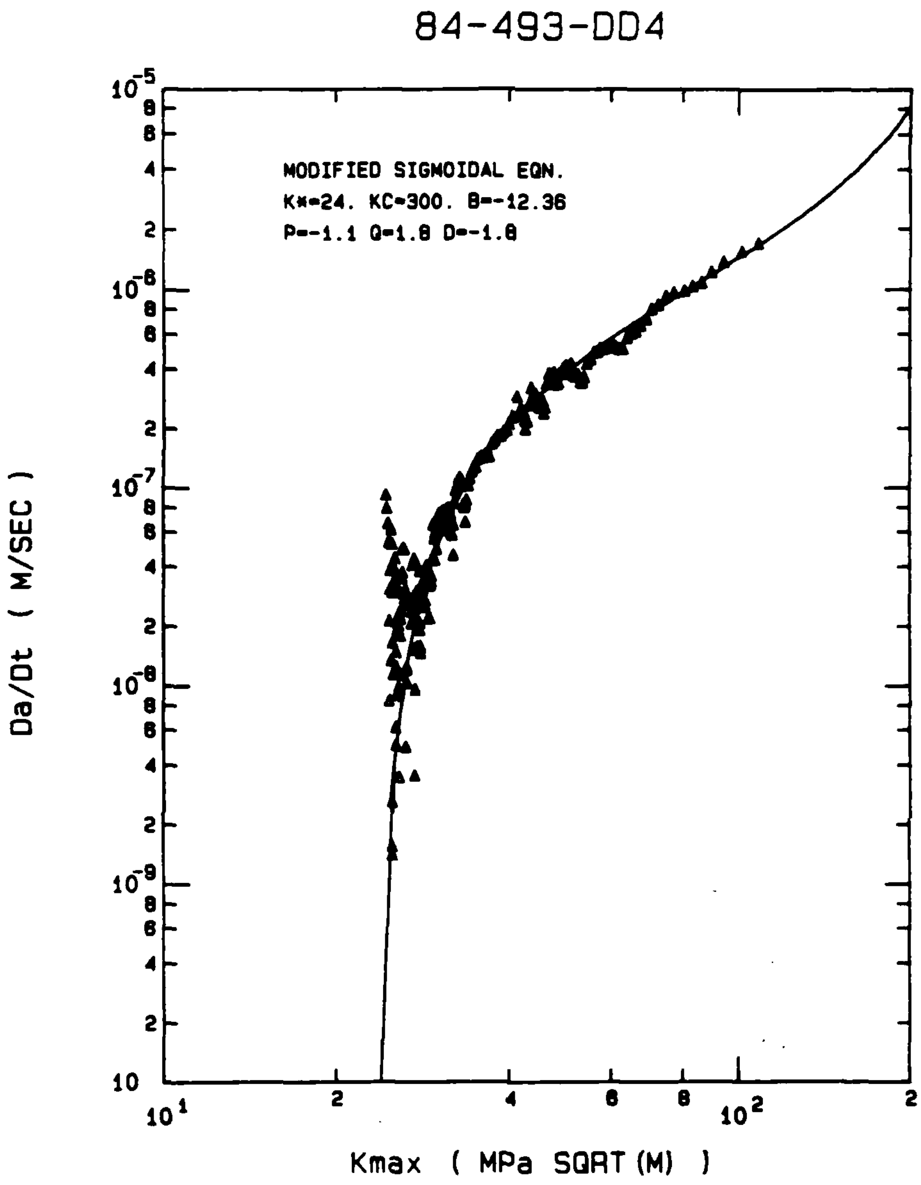


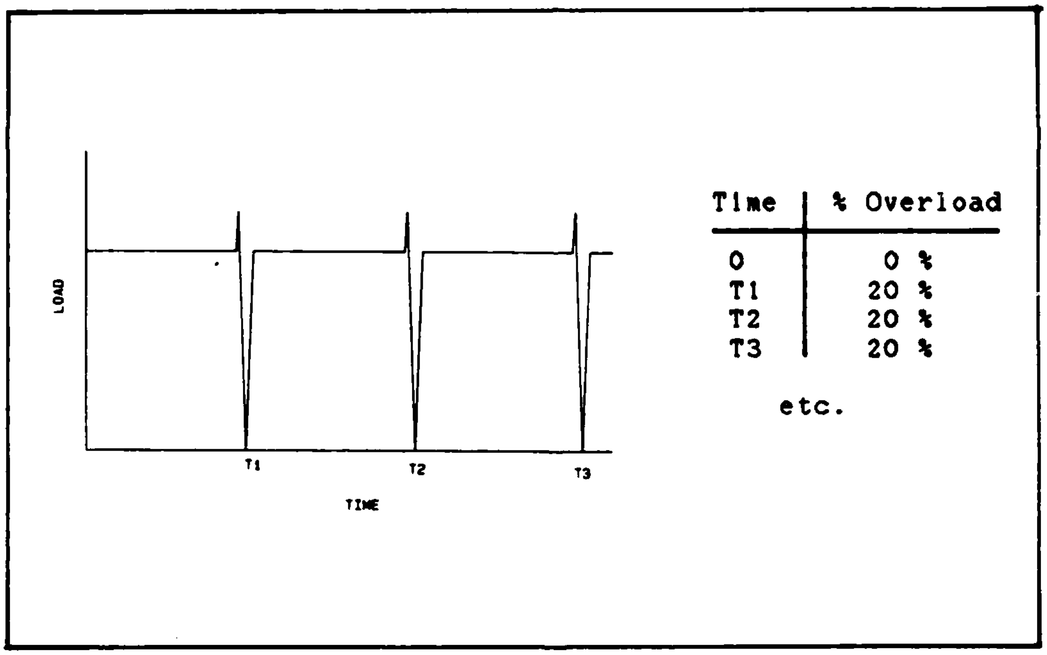
Figure 6 MSE Model Fitted to Sustained-Load Baseline Data.

The accuracy of the integration was controlled in two ways. First, the interval of total crack length ($a_f - a_i$) was divided into a number of increments (Δa). The size of each increment was based on the initial stress intensity level at the beginning of the interval and was determined as follows. Values of $(da/dt)_{MSE}$ were calculated at various K-levels. These values were multiplied by 100 seconds to obtain the crack length interval (Δa_{100}) which produces 100 seconds of sustained load growth. Linear functions were used to represent Δa_{100} as a function of K, in a manner that insured the growth time interval was always less than 100 seconds. The effect of this first step was to use small Δa intervals at lower K values and larger Δa intervals as K increased, thus limiting the total time required for the crack to grow through any interval to less than 100 seconds, for any value of crack length or K.

Each Δa interval was then numerically integrated using Simpson's rule. The numerical integration errors were minimized by testing the Simpson's integration subroutine for convergence. This was accomplished by integrating each crack length interval at least twice. The program started by using two subintervals within each Δa interval. This interval was then integrated again doubling the number of subintervals. The difference of the values of the time calculated to grow through the interval using these two integrations was compared with a convergence value; this value was calculated by multiplying a tolerance, set at 0.0001, by the time calculated to grow through the crack interval. Since this time increment was always near 100 seconds, the convergence value corresponded to approximately 0.01 seconds difference between the calculated times using the two successive integrations. If the difference exceeded 0.01 seconds, another integration was performed, with the number of subintervals being doubled again. This procedure was repeated until the convergence criterion was satisfied. Assuming that each Δa interval resulted in an error of 0.01 seconds, the cumulative error for the total time predictions is no greater than 10 seconds based on a maximum number of 1000 intervals corresponding to a total test time of approximately 10^5 seconds.

The numerical integration scheme was applied to a loading history which involved a constant sustained-load with periodic overloads. The input load history for each specimen specified the times at which overloads were applied and the percentage of overload. An example of how the load history was entered into the program is shown in figure 7.

The program integrates repeated Δa intervals, equal to approximately 100 seconds of sustained load time, until the total time exceeded the time when the next overload was applied. At this point, the time when the last Δa interval started and the time it took to grow the last Δa interval were known. Using a linear interpolation, the length of the last Δa interval was reduced so that the total growth time equaled the time when the overload was applied. At this point, the retardation effect due to the overload was added in the model. This was accomplished by calculating a reduced stress intensity factor, K_{eff}, defined by equation (8). In this equation, the modeling parameter β was related to the overload plastic zone size, while the parameter α was defined by fitting experimental data. During the retarded growth K_{eff} was substituted for K in equation (18) until the crack grew through the overload plastic zone.



Time	% Overload
0	0 %
T1	20 %
T2	20 %
T3	20 %

etc.

Figure 7 Overload Program Load History Input Example.

Harms [7] noted that each time an overload cycle was applied an apparent jump in crack length occurred. This same jump phenomenon was also noted by Larsen and Nicholas [12] in their study of crack-growth transients at elevated temperature. The amount of crack jump seemed to correlate with the K level at overload application. However, Harms's data for jump versus K level, shown in figure 8, contained a large amount of scatter making it difficult to fit a functional relationship. Harms therefore assumed a constant value of 0.381 mm jump in crack length at each overload. In an attempt to improve the estimation of the jump function, several other functional relationships were investigated.

First, a function relating the amount of jump to the K level at overload application, normalized to the threshold K^* value of 24 MPa $m^{1/2}$, was tried. This function takes the form of

$$\text{Jump} = 0.2 * (K/K^*) \ \text{mm} \qquad (20)$$

and is labeled curve 1 in figure 8. The second function used the Log of the K level at overload application, normalized to the threshold K^* value of 24 MPa $m^{1/2}$. The resulting function takes the form of

$$\text{Jump} = 1.25 * \text{Log}(K/K^*) \ \text{mm} \qquad (21)$$

and is labeled curve 2 in figure 8. Finally, a linear function with an initial jump of 0.381 mm at threshold increasing to .508 mm jump near the critical stress intensity was tried. This function is labeled curve 3 in figure 8. Also shown in this figure is the constant 0.381 mm jump used by Harms, labeled curve 4.

The proof test, conducted by Harms, was analyzed using each jump function to predict the increment in crack length caused by the overload cycle. The resulting predictions are shown in figure 9. Curves 1 and 2 correspond to jump functions in equations 20 and 21, respectively these functions underestimated the jump at low K values. The resulting predictions had significant delay times. Curve 3 corresponded to the linearly increasing jump function, and was the most exact prediction of the total time to failure for this test. The constant jump of 0.381 mm, labeled curve 4, predicted the total time to failure to within 4 percent. Since the constant jump used by Harms gave predictions within normal test scatter, it was decided to use a constant jump of 0.381mm for all further calculations. This also allowed a direct comparison of Harms's work with the other retardation models.

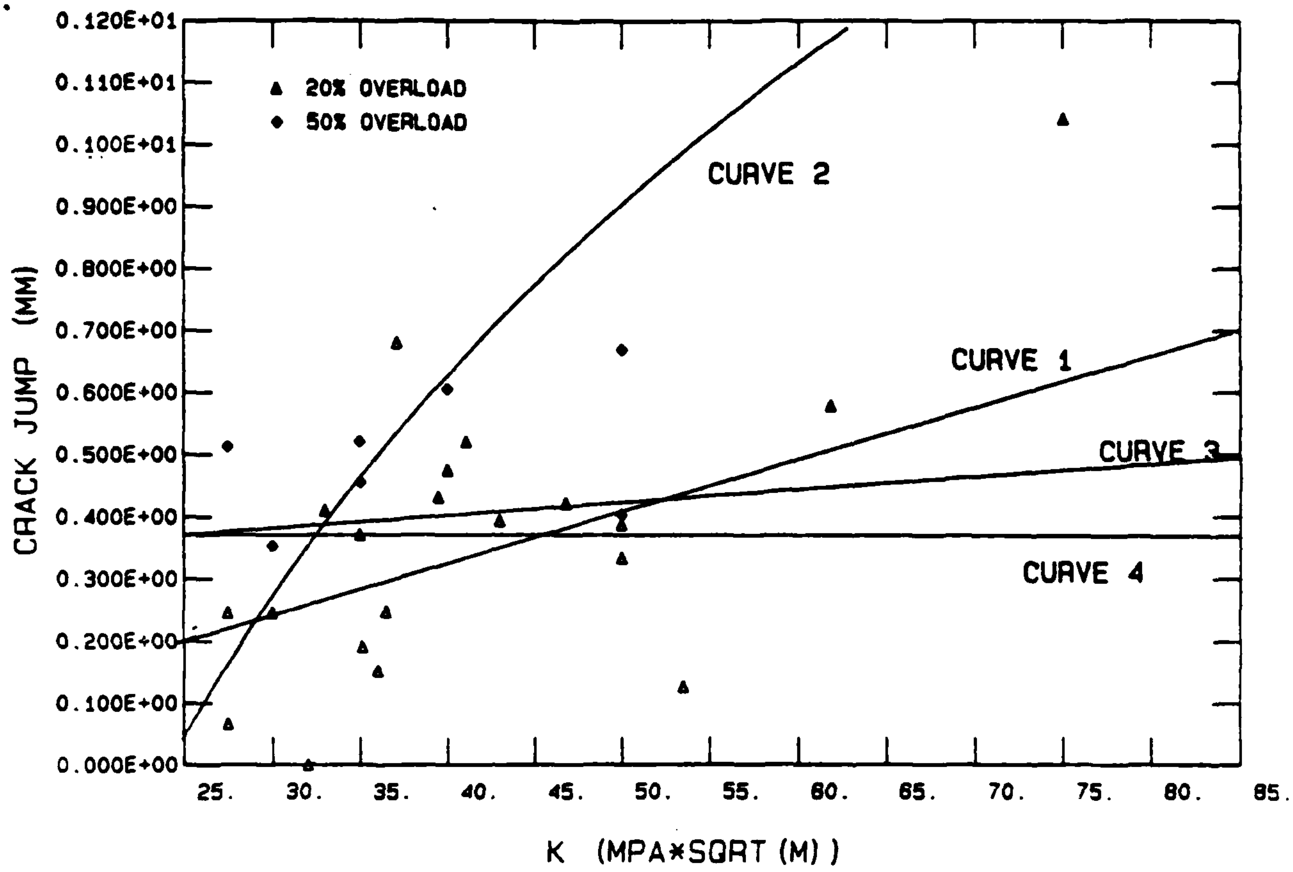


Figure 8 Crack Jumps from Overload Cycles at Various K Levels.

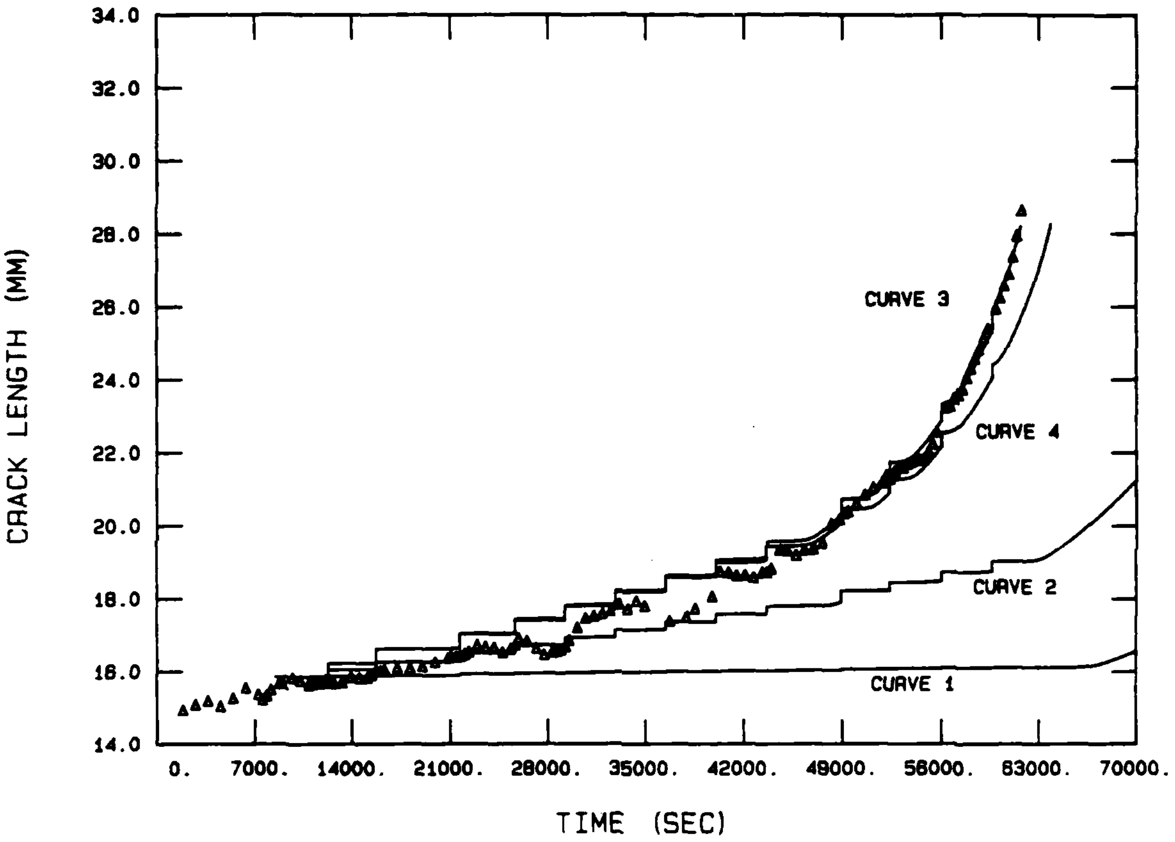


Figure 9 Overload Model Predictions to Proof Test 84-507 Using Various Jump Functions.

CRACKS Program

This program was developed by R. M. Engle [4] to predict crack growth in airframe applications where high frequency, low temperature spectrum loading occurs. The objective here was to modify the original program so that it could be used to analyze high temperature sustained loading with periodic overloads. A technique was developed for converting sustained loading to equivalent fatigue cycles. This technique and all required programing changes made within CRACKS are described next.

The process of converting sustained load creep crack growth to equivalent fatigue cycles was achieved by setting da/dn equal to da/dt at equivalent ΔK and K_{max} stress intensity levels. A one-to-one correspondence would equate one fatigue cycle to one second of sustained load. Since CRACKS can only use one crack growth rate equation, a method of representing both the rate of growth due to sustained loading and overload fatigue cycles with the same equation was needed. The growth rate due to an overload cycle was approximated using previously generated test data [13] for Inconel 718 at 650 C, with an R ratio of 0.1 and frequency of 0.01 Hz. This frequency is approximately that of the single overload cycle in the experimental part of the investigation. This data is labeled 83266G test data and shown in figure 10. Also shown in the figure is the baseline

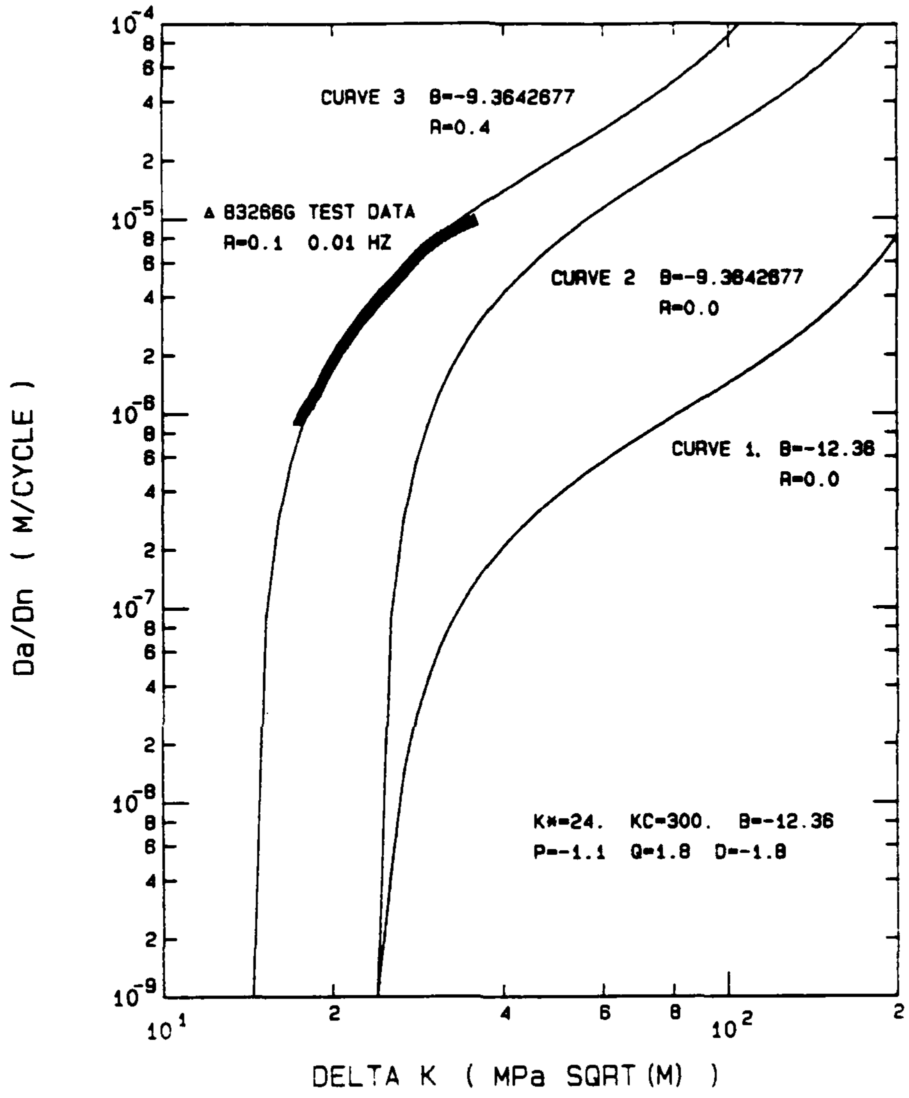


Figure 10 MSE Curve Shift Using the B Parameter and R Ratio.

sustained-load sigmoidal curve, developed by Harms, and labeled curve 1. The sustained-load growth rate da/dt (m/sec) given by curve 1 equals the cyclic growth rate da/dn (m/cycle). Thus, one second of sustained-load is equal to one fatigue cycle. In order to have the same curve represent both the sustained loading and overload cycles a vertical and horizontal shift of curve 1 was needed. Refering to the sustained load sigmoidal equation below

$$da/dt=[\exp(B)]*[K/K^{*}]^{P}*[\ln(K/K^{*})]^{Q}*[\ln(K_{c}/K)]^{D} \quad (22)$$

the parameter B can be used to absorb a rate multiplication constant into the equation. Using a trial and error procedure, a factor of 20 was found to vertically shift the curve to a position where an additional horizontal shift placed the curve on the crack growth rate data for overload cycles. The new B value was found by solving the following equation for B

$$\exp B = 20 \exp B_{old} \quad (23)$$

The new value for B and the associated new sigmoidal curve is labeled curve 2 in figure 10. The cyclic growth da/dn (m/cycle) now equals 20 times the sustained-load growth rate da/dt (m/sec). Thus, 20 seconds of sustained-load is equal to one fatigue cycle. The horizontal shift was accomplished using an R ratio shift. The cyclic sigmoidal equation shown below

$$da/dn=[\exp(B)]*[\Delta K/\Delta K^*]^P*[\ln(\Delta K/\Delta K^*)]^Q*[\ln(\Delta K_c/\Delta K)]^D \quad (24)$$

can be modified for R ratio effects by replacing ΔK with $\Delta K*(1 - R)$, ΔK^* with $\Delta K^**(1 - R)$, and ΔK_c with $\Delta K_c*(1 - R)$. Again, using trial and error,it was found that an R ratio of 0.4 produced the desired horizontal shift. The resulting sigmoidal equation with fixed a value for $\Delta K^* = \Delta K^**(1 - 0.4)$ and $\Delta K_c = \Delta K_c*(1 - 0.4)$ is labeled curve 3 in figure 10. Both the sustained load and overload cycle can be modeled by varying the R ratio to represent the different cycles. The sustained loading is modeled using 1 cycle equals 20 seconds of sustained loading at an R ratio of 0.4. The overload cycle is modeled as 1 cycle equals 1 overload cycle at an R ratio of 0.0. A graphical representation of these cycles is shown in figure 11. Since $\Delta K = \Delta K*(1 - R)$ and R = 0.4 for the sustained loading, it is easily seen that the R ratio dependence cancels out of equation (24) for sustained loading. The sustained load crack growth rate then in essence is defined by curve 2 in figure 10. Remembering that the threshold and critical stress intensity factors are fixed at values corresponding to an R ratio of 0.4 and the overload cycle is modeled with an R ratio of 0.0, the overload cycle crack growth rate is calculated using curve 3 in figure 10 with $\Delta K = \Delta K*(1 - 0.0)$ or just ΔK.

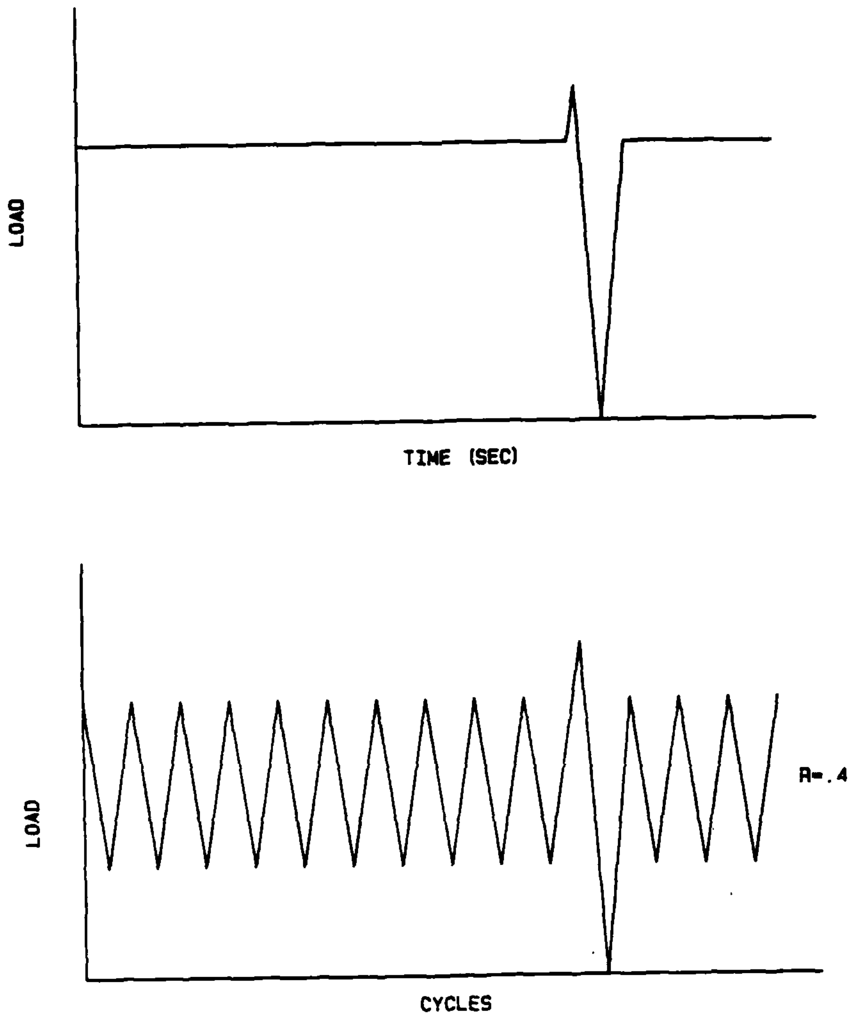


Figure 11 Graphical Representation of Equivalent Sustained-Load Fatigue Cycles.

Implementation of the method developed to change sustained loading into equivalent fatigue cycles required several changes to the CRACKS program. The source listing of the program used in this study is contained in appendix 3. Before describing the detailed changes within the program, a brief description of the overall program will be presented.

The CRACKS program consists of twenty two routines of which sixteen are basic to crack growth calculations and six others are used to implement the retardation models. Detailed descriptions of each routine is contained in the CRACKS manual [4]. The overall supervisory routine CRACKS4 calls each of the other subroutines as needed during the calculations. Only the major changes made within each subroutine will be described, although each change usually required changing other subroutines that used the same common blocks.

The first change was to include the sigmoidal crack growth rate equation in the RATE subroutine. This required changing the input and output subroutines to read and write the sigmoidal coefficients. Also, equation (24) was programed in the RATE subroutine for use whenever the crack growth rate was required by the main CRACKS4 routine.

The second change was to include the ASTM compact tension stress intensity solution, given by equation (17), in the BETA subroutine. The CRACKS program calculates all K values using the equation

$$K = \sigma Y \sqrt{\pi a} \qquad (25)$$

where K = Stress intensity level

σ = Stress or load

Y = Variable defining case solution

a = Current crack length

The variable Y is used to define the case solution for the type geometry being analyzed. Y was set equal to equation (17) divided by $\sigma\sqrt{\pi a}$. Whenever the stress intensity value for a given crack length was needed, the subroutine K was called. This subroutine called the BETA subroutine where a new value for Y was calculated for substitution into equation (25).

The final change made in CRACKS affected the numerical integration method used in the program. CRACKS was written to handle a very large number of fatigue cycles. Crack growth due to large spectra of cycles are calculated using a linear approximation technique in order to save computation time. The basis for the approximation is the assumption that the damage parameters remain constant over some small increment of crack growth Δa. Engle [14] found the linear approximation in CRACKS to be an excellent balance between accuracy and computational efficiency for very large spectra. However, when smaller constant amplitude spectra were analyzed, the program was more efficient using a cycle-by-cycle Runge-Kutta numerical integration method [14]. During the check-out phase of the CRACKS program, it was found that the linear approximation did not provide the same accuracy as the Overload program for constant sustained loading. Therefore, the program was changed to eliminate the linear approximation and use the Runge-Kutta cycle-by-cycle integration method. With this change in place both the Overload and CRACKS programs predicted exactly the same results for constant sustained-load growth containing no overloads.

Results from the CRACKS program were obtained in terms of equivalent fatigue cycles of growth. This required a separate Fortran program to convert cycles back into sustained load time by multiplying the equivalent cycles by 20 seconds per cycle.

Both the Overload and CRACKS programs were now capable of predicting sustained load crack growth with periodic overloads. The retardation effect overload cycles produced was estimated using the Overload model in the Overload program and the Wheeler and Willenborg models in the CRACKS program. The next step in developing the programs was to compare the analytical predictions of each model to experimental test data.

IV. Application of Retardation Models

Each retardation model was applied to typical test segments using the computer programs described in section III to predict how sustained load crack growth was affected by overload cycles. Six segments of crack growth data with average delay times were selected from Harms's work for analysis. The segments included both 20 % and 50 % overloads applied at low (30 MPa $m^{1/2}$), medium (40 MPa $m^{1/2}$) and high (50 MPa $m^{1/2}$) stress intensity levels. The test data for each segment, shown in figures 12 through 17, start at an initial crack length which already includes the crack length jump produced by the overload cycle. Crack growth predictions using the Overload, Wheeler, and Willenborg retardation models were computed for each segment to provide a comparison between the predictive capabilities of each model. Several observations were made on the flexibility of each retardation model to predict the test data. Discussion of each of the observations and their effects follows.

The Overload model's flexibility to match test data is contained in the α parameter in equation (8). This parameter controls the delay time before normal crack growth resumes after an overload. Harms used his test data to develop expressions for α as a function of stress intensity level for 20 % and 50 % overload ratios. These expressions, given by equations (15) and (16), were used in the Overload model. Thus all the model variables were predefined before the test segment were analyzed. The predictions for each of the six segments are shown in figures 12 through 17.

The Wheeler model had flexibility to fit test data by changing the shaping exponent m. The shaping exponent m is an empirical parameter dependent upon material and stress history [15]. The value of m for fatigue cycling generally ranges from 1.0 to 3.5 . During analysis of the six test segments the shaping exponent m was treated as a variable to fit the Wheeler model to the test data. Figure 18 shows how the best fit value of m was found by trial and error for a typical test segment. A constant value of m equal to 6.0 accurately predicted the 20 % overload segments. The best fit value of m for the 50 % overload segments was related to the stress intensity level at overload application and varied between 6.0 and 3.5 . This relationship between m and K is shown in figure 19. In general, the shaping exponent decreases as K increases at higher overload ratios. A similar trend was seen in the α parameter which is used to fit experimental data in the Overload model. Accurate predictions of retardation at higher overload ratios depends upon relating α or m to the stress intensity at overload application. At lower overload ratios α was still related to the stress intensity level but a constant value of m was found to adequately predict the retardation affect. The resulting Wheeler predictions using the best fit values of m are shown in figures 12 through 17.

The Willenborg model does not have a parameter, like the Overload and Wheeler models, for use in fitting test data. Instead the model accounts for retardation by reducing the equivalent sustained load fatigue cycle of

$$\Delta K = K_{max} - K_{min} \qquad (26)$$

with $R = K_{min}/K_{max} = .4$

to an effective value calculated by substituting $[K_{max}, K_{min}]_{eff}$ from equation (7) into equation (26) to get

$$\Delta K_{eff} = [\Delta K_{max}]_{eff} - [\Delta K_{min}]_{eff} \qquad (27)$$

with $R_{eff} = [K_{min}]_{eff} / [K_{max}]_{eff}$

Application of the Willenborg model to the test segments showed that R_{eff} did not equal the equivalent fatigue cycle R ratio of 0.4. This is due to the truncation of the minimum K_{eff} value at zero and the reduction of both K_{max} and K_{min} by K_{red} as defined in equation (7). Therefore, the model would not predict the same retardation effect if the equivalent sustained load cycles were modeled at different R ratios. The resulting predictions using the Willenborg model with the equivalent fatigue cycle modeled with an R ratio of 0.4 are shown in figures 12 through 17. Since the Willenborg retardation model was dependent on the modeling technique used to represent the sustained load time it was eliminated from further consideration in this study.

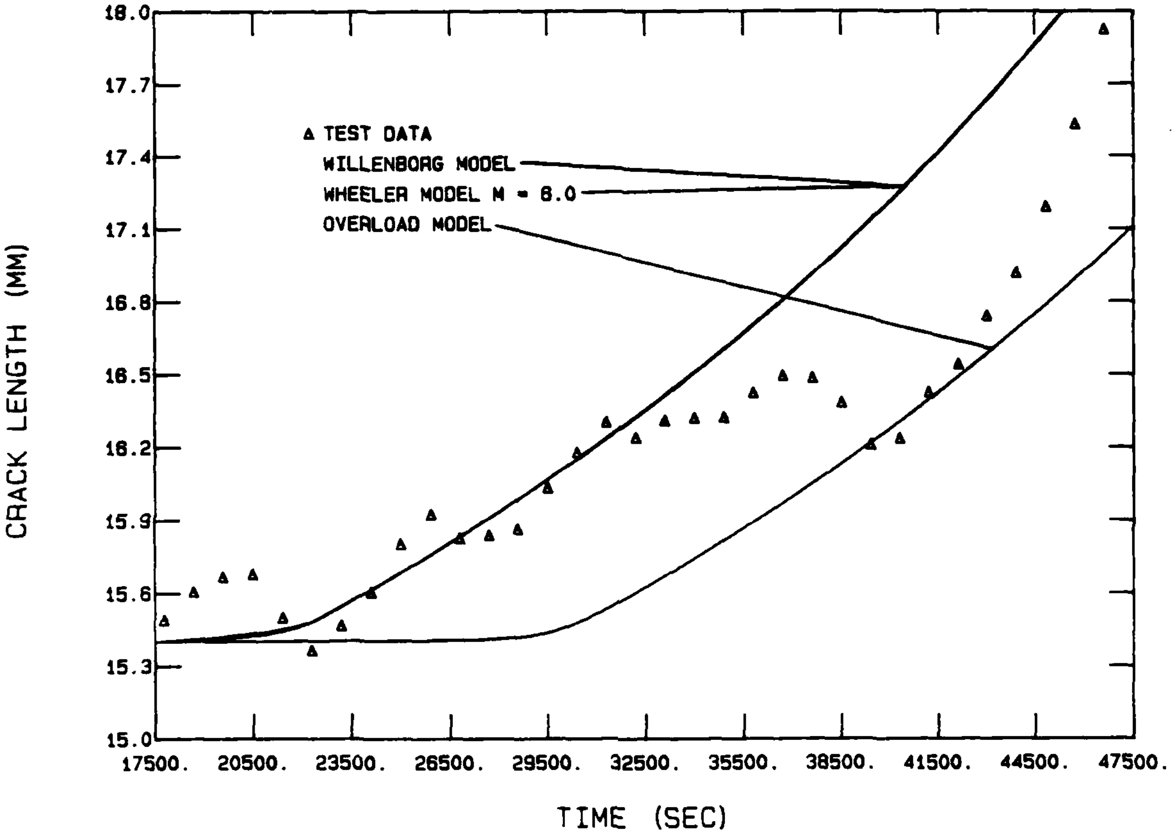


Figure 12 Crack Length versus Time for

20 Percent Overload at Low K.

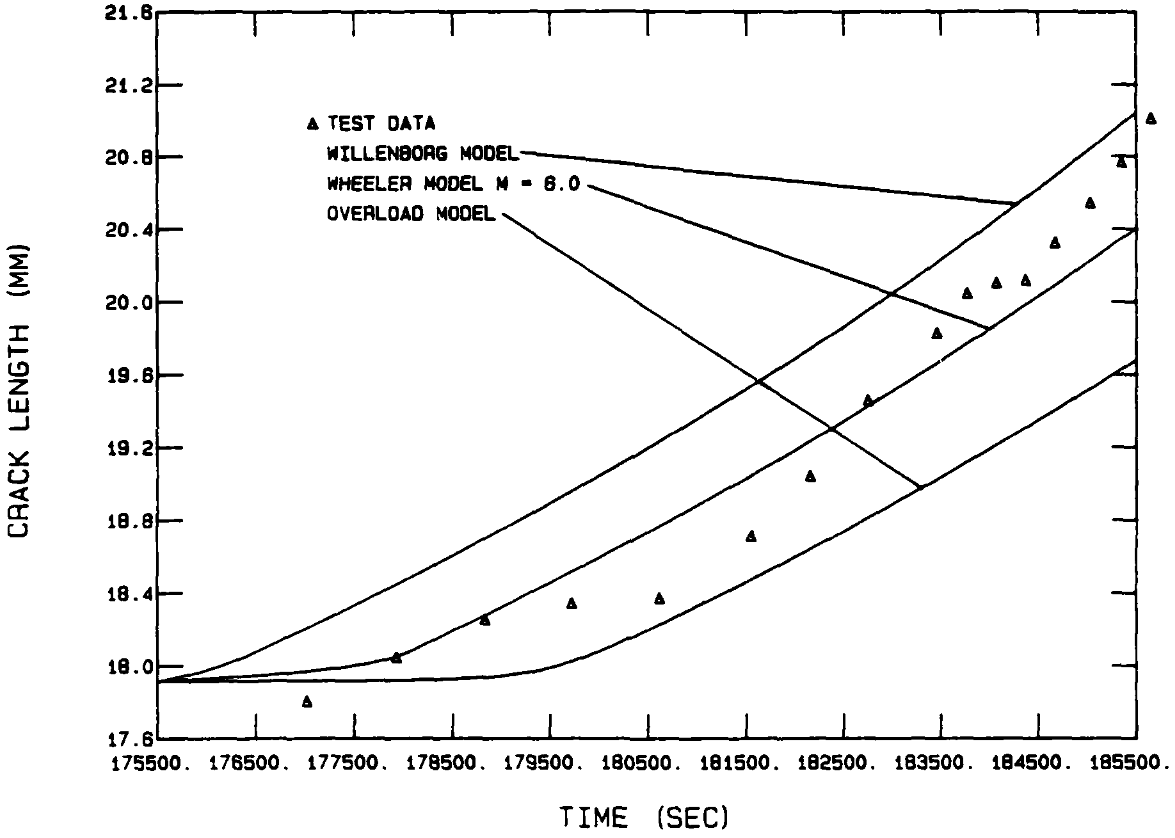


Figure 13 Crack Length versus Time for

20 Percent Overload at Medium K.

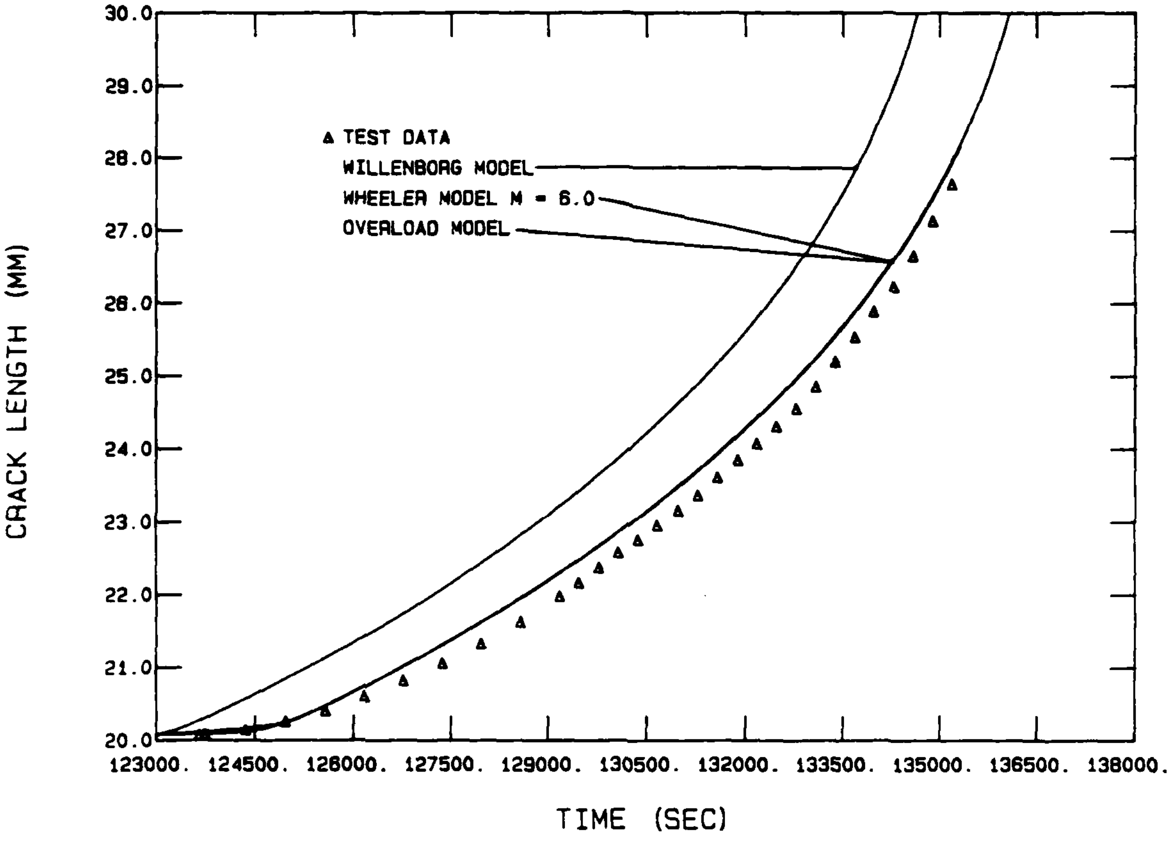


Figure 14 Crack Length versus Time for

20 Percent Overload at High K.

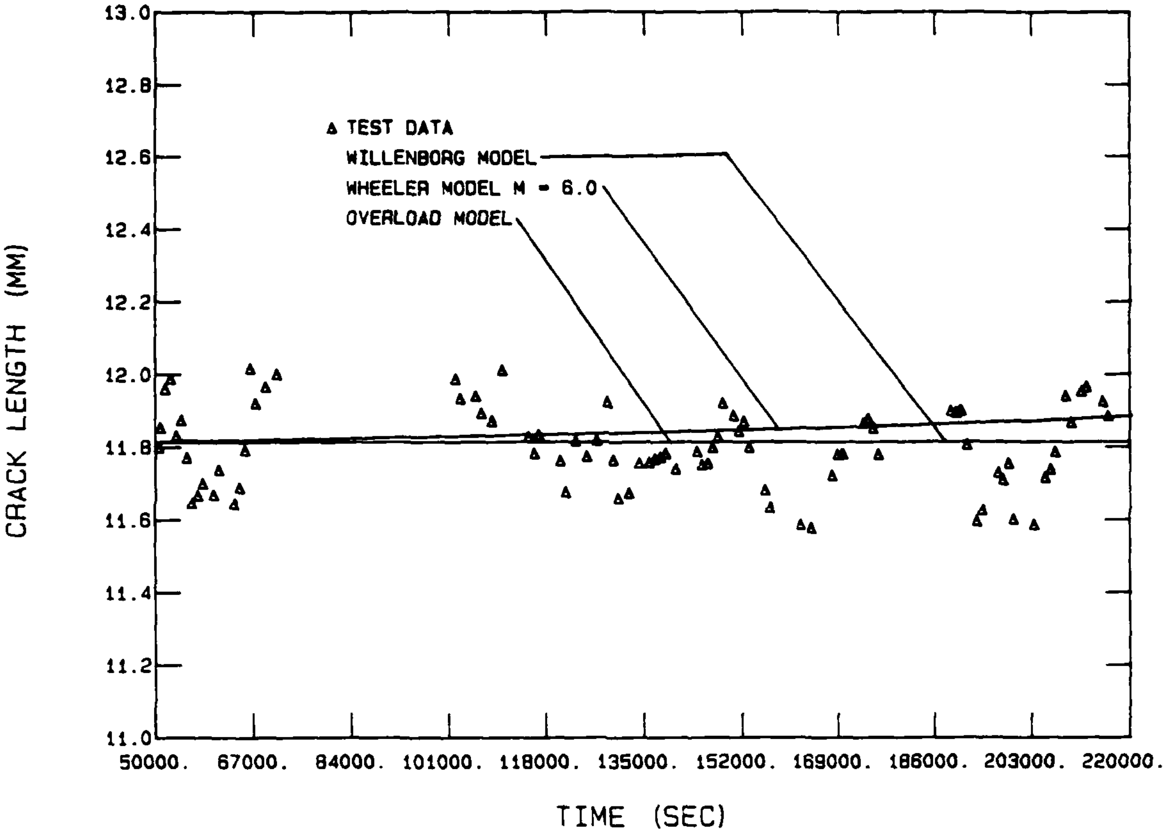


Figure 15 Crack Length versus Time for

50 Percent Overload at Low K.

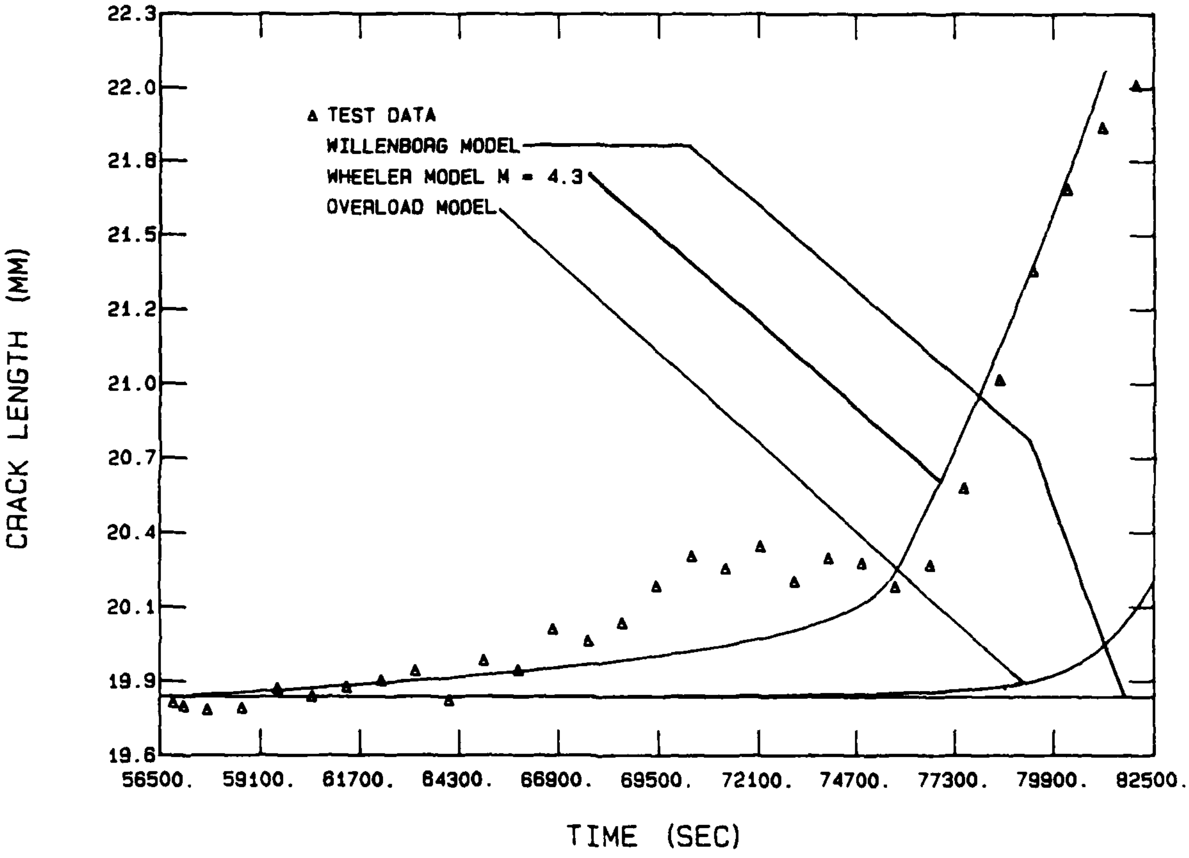


Figure 16 Crack Length versus Time for

50 Percent Overload at Medium K.

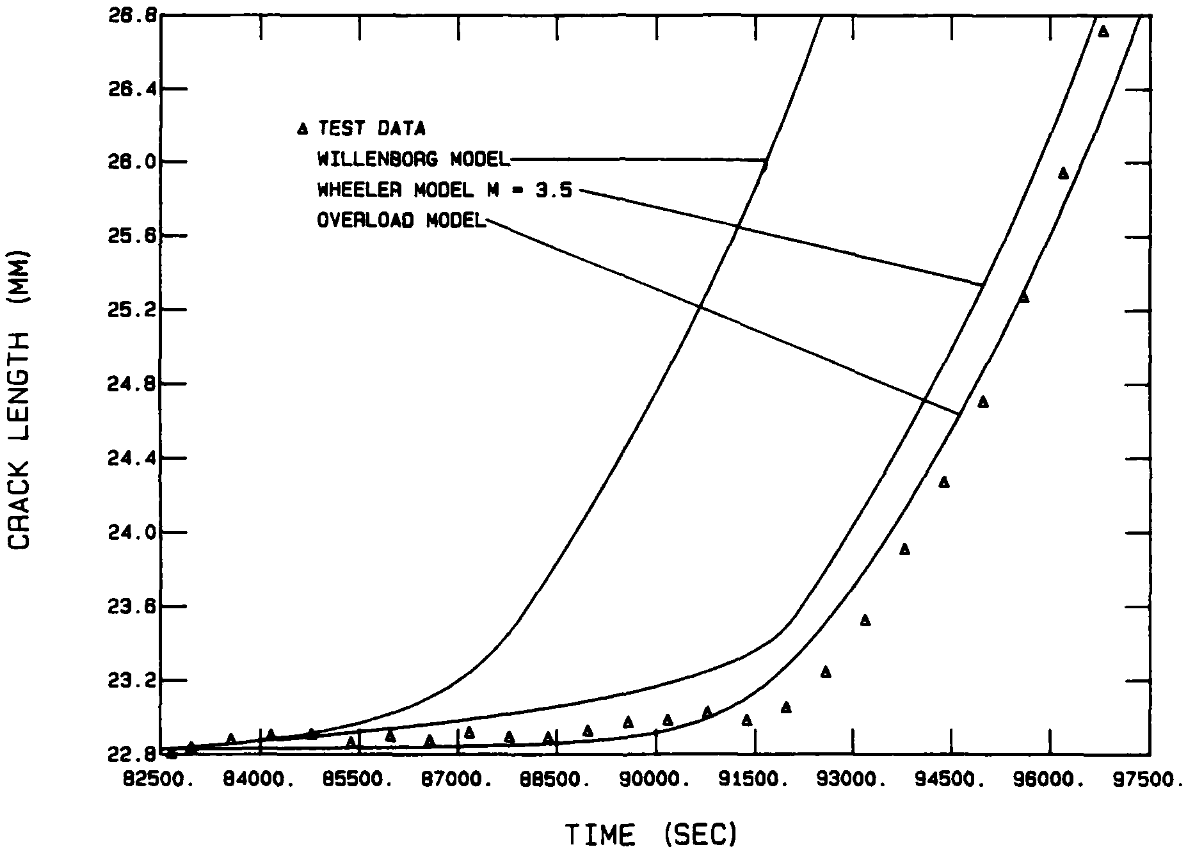


Figure 17 Crack Length versus Time for 50 Percent Overload at High K.

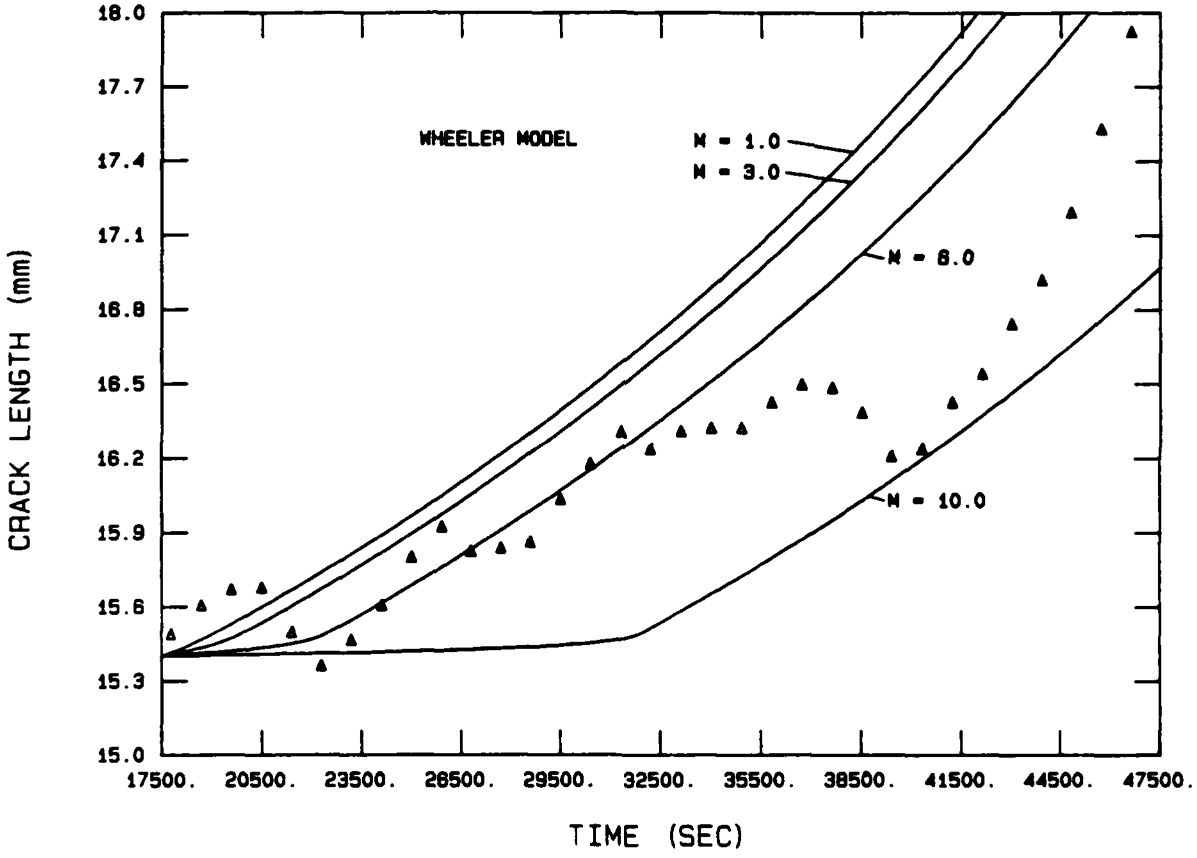


Figure 18 Best Fit Shaping Exponent (m)

for 20 Percent Overload at Low K.

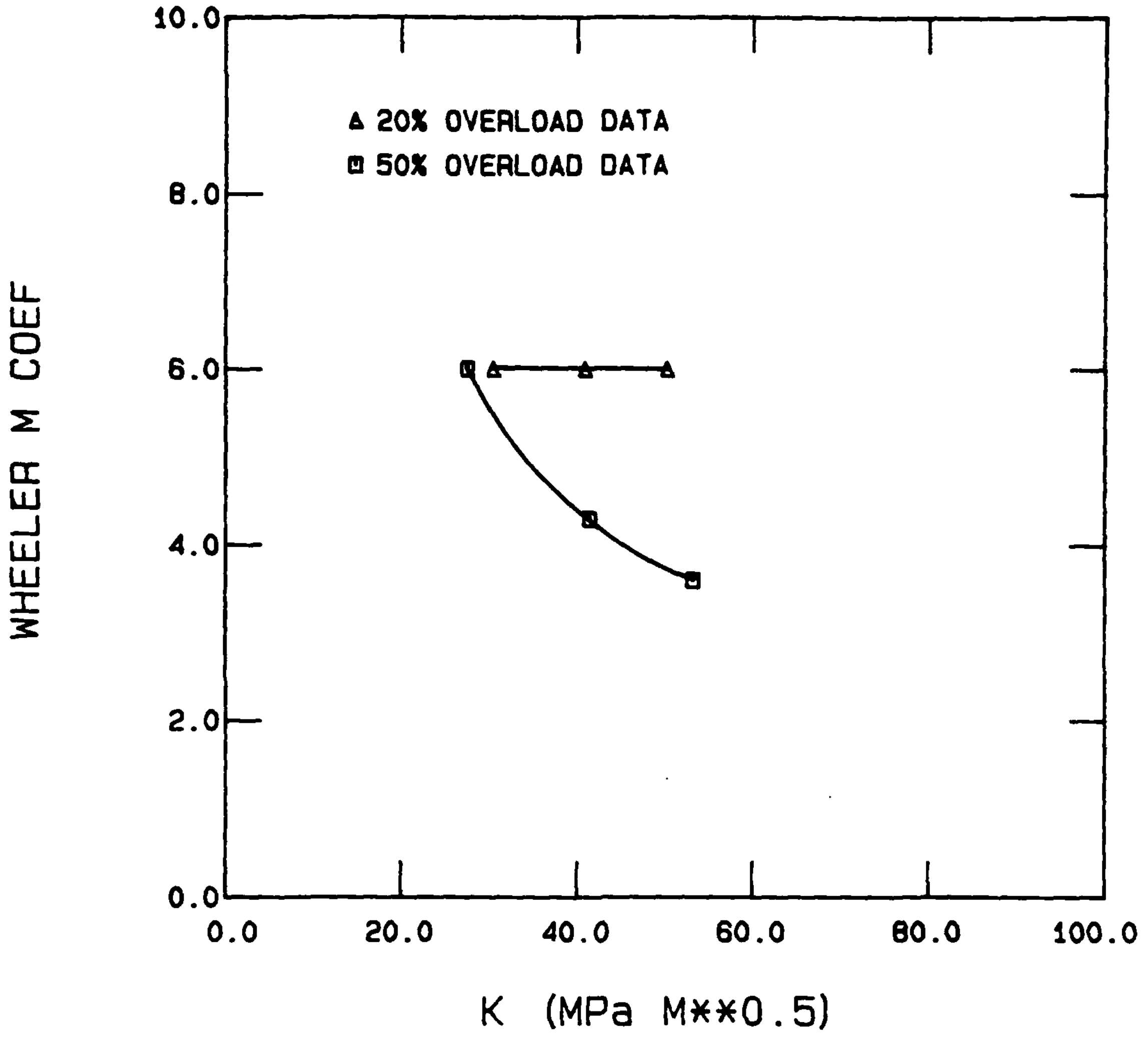


Figure 19 Best Fit Shaping Exponent (m) versus K.

V. Proof Test Experiments

Several proof tests were conducted to provide additional test data to verify the prediction capability of the retardation models. A description of the test apparatus and crack measurement procedures used during testing follows.

Test Apparatus

The experimental data for the proof tests was gathered using a semi-automated creep test system employing electric potential drop and optical readings to monitor crack length. The Air Force Wright Aeronautical Laboratories, Materials Laboratory, Wright-Patterson Air Force Base, Oh. provided the facilities and equipment to conduct the testing. The test system schematic is shown in figure 20. The test setup included the following components:

1. 12,000 lb Swedish creep test frame
2. Resistance heated furnace with power controlers
3. Tektronix 4051 microcomputer
4. Daytronic 9000 signal processor
5. Hewlett-Packard 3478A IEEE-488 digital voltmeter
6. Two Gaertner traveling microscopes
7. Current supply source

The 12,000 lb.-capacity Swedish creep frame was used to load the specimens. The frame is constructed using a lever and fulcrum principle. The weights were suspended at the end of a 20 to 1 lever arm. The other end reacts to this mechanical advantage with a load line containing the specimen. Dynamic loading of the specimen is avoided by the use of a hydraulic ram which can support a fraction or all of the suspended weight. As pressure is added or removed from the ram, it removes or adds the load to the specimen in a smooth manner. Periodic overloads were applied by unloading the specimen, adding the calculated overload weight to the end of the lever arm, and reloading the specimen. Removing overloads was done using the same procedure except for removing the weights.

The compact tension specimens were mounted in the load line using load rods with Inconel 718 clevises and holding pins. An electric potential technique, to be discussed below, was used to measure crack length. To accommodate this system, the specimen was electrically isolated from the creep frame by using insulated sleeves and pins to connect the load transfer rods. A load cell to measure the applied load to the specimen was included in the load line. The load cell readings were used to ensure the hydraulic ram released the entire load to the specimen. The specimen's configuration and nominal dimensions are shown in figure 21. In addition, the individual test specimen's dimensions and the applied loads are listed in table 1.

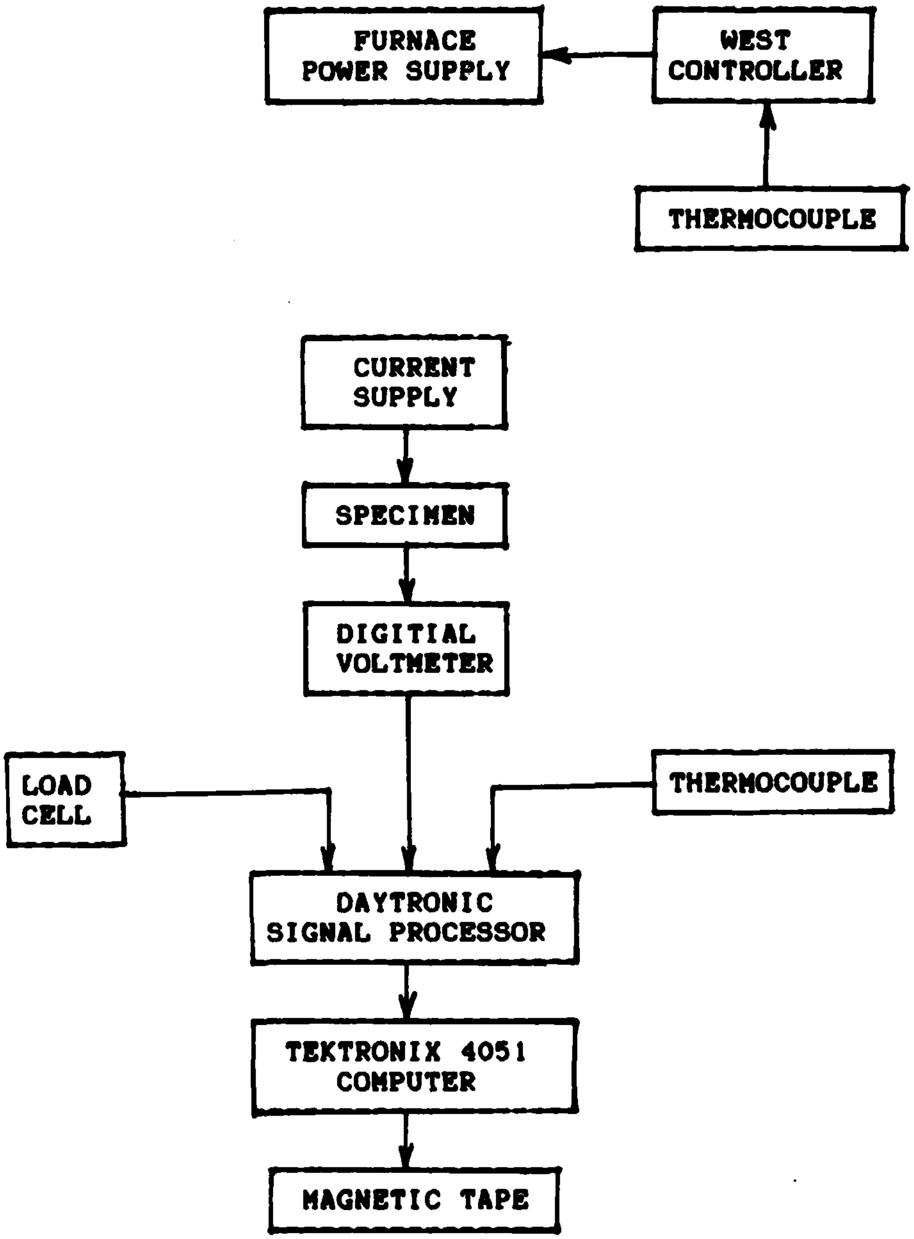
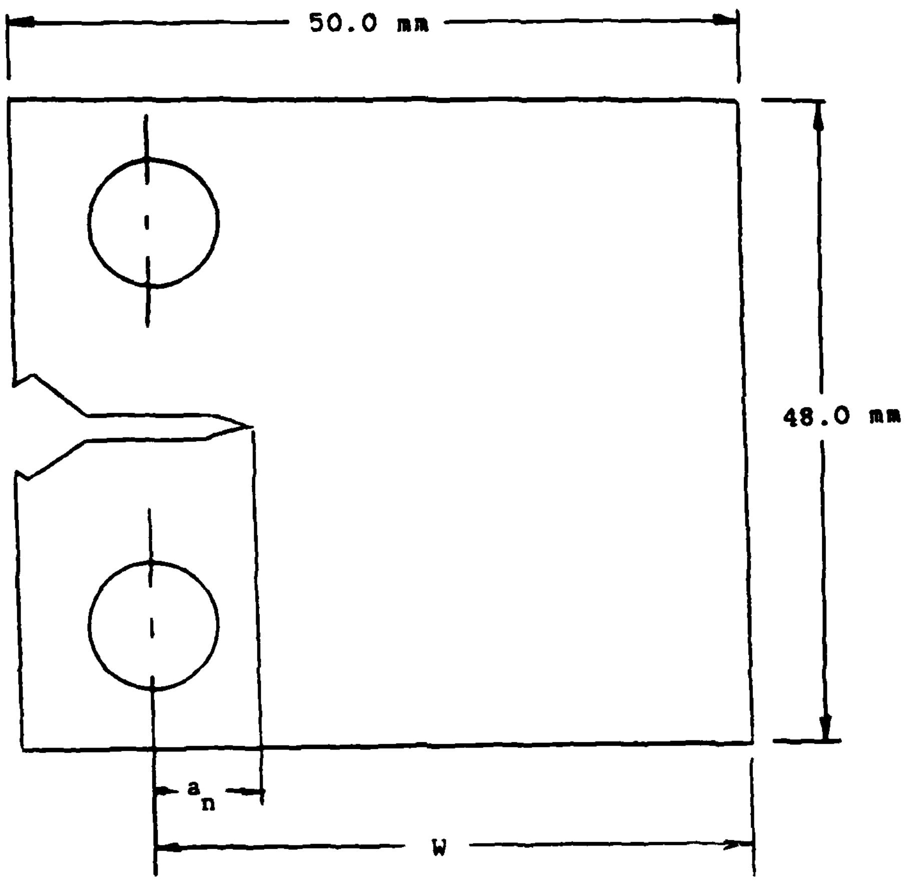


Figure 20 Test System Schematic.



a_n = 7.0 mm
W = 40.0 mm
thickness (B) = 10.0 mm

Figure 21 Compact Tension Specimen.

Specimen	a_i (mm)	Width (mm)	Thickness (mm)	Load (KN)
84-502	10.381	39.969	10.008	12.188
84-503	10.665	40.008	10.033	12.023
84-504	10.267	40.018	10.008	12.677

Table 1 Specimen Dimensions and Applied Load.

The two piece resistance furnace was mounted on the creep frame as shown in figure 22. The oven was closed around the specimen and sealed with a flameproof wadding. The wadding material also served to insulate the leads for the electric potential system from the oven and frame. The furnace was constructed with four independently controlled power zones. The power to each zone is controlled by a time-proportioning West controller. The controller used a thermostat feedback loop to hold the oven's temperature constant. Two K-type, chromel-alumel, thermocouples were spot welded to the specimens and are shown in figure 23. One was used by the controller and the other served as a backup (connected to the Daytronic controller for periodic monitoring).

The Tektronix 4051 microcomputer was used as the data-acquisition system for the tests. The Tektronix was programed to take electric potential readings at predetermined time intervals. The time input signal sent to the Tektronix was provided by the Daytronic controller.

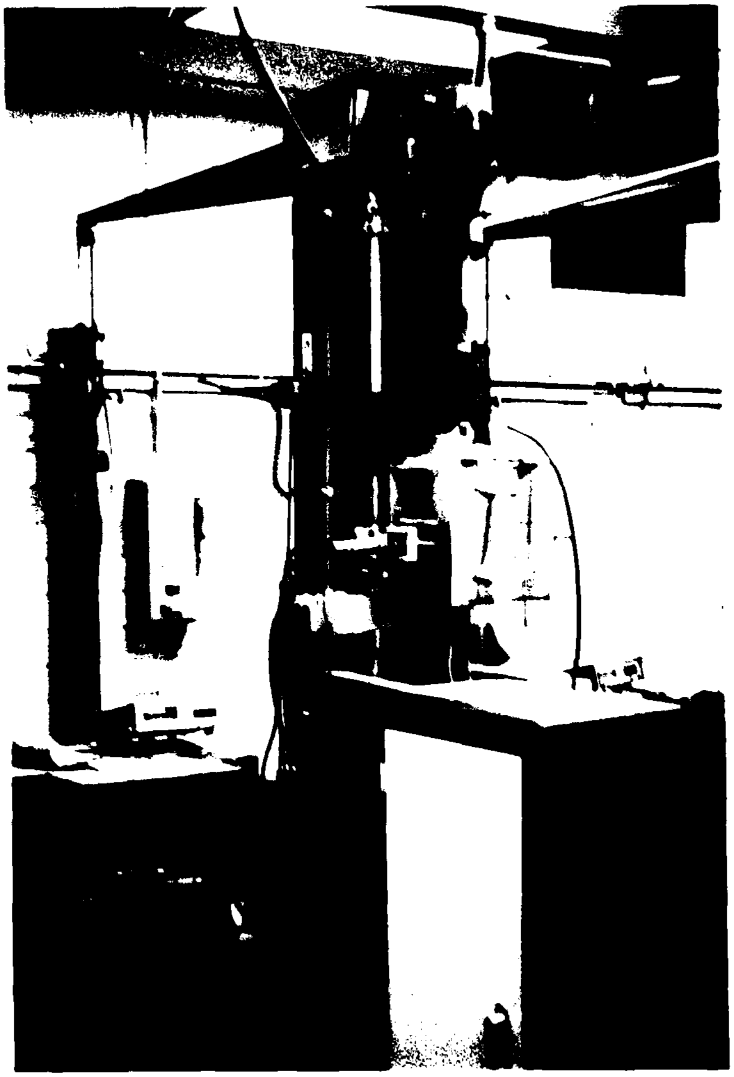


Figure 22 Creep Frame Photograph.

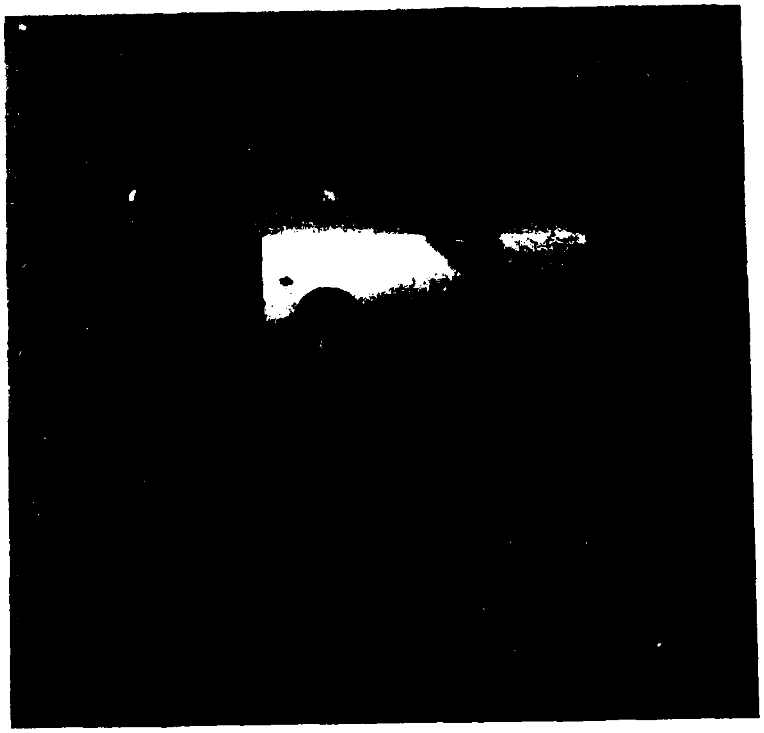


Figure 23 Instrumented Specimen Photograph.

Similarly, the Hewlett Packard 3478A IEEE-488 programable voltmeter provided the voltage reading when requested by the Tektronix's program. The two received signals were then recorded on magnetic tape. This information was later transferred to the host PDP 11/24 computer for use in the data reduction procedures to obtain crack length from voltage. In addition, optical crack length measurements were obtained through the viewing ports on each side of the resistance furnace using two Gaertner traveling microscopes. The crack length measurement was displayed on a digital readout to an accuracy of 0.0254 mm. However, due to optical problems in determining exactly where the crack ended, measurements were reproducible to an accuracy on the order of 0.05 mm. The optical crack length measurements were manually recorded periodically along with time and the corresponding voltage readings from the electric potential system in order to provide data for later correlation of crack length to voltage.

Crack Measurement Procedure

The proof experiments utilized a D.C. Electric Potential (EP) measurement system augmented with optical readings to monitor crack length. The EP system is based on the fact that when a current is passed through a conducting body an electric field is generated. The field shape and intensity depend upon such factors as applied current, geometry shape and material properties. The EP system relies on relating changes in the EP field at the output leads to changes in geometry due to crack growth.

One of the instrumented specimens is shown in figure 23. The input and output leads were welded to the specimen at the locations shown in figure 24. The input leads, made of Inconel 718, were connected to a constant 10.0 amp current source. The output voltage was measured from nichrome wire leads spot welded on the specimen front surface. Nichrome wire was used for the output leads due to its superior oxidation resistance at elevated temperature. However, joining two dissimilar materials produces a thermocouple effect as the joint temperature is changed. This thermal voltage adds algebraically to the voltage generated due to the resistance in the specimen. It should be noted that the thermal voltage is present even when the applied input current is removed from the specimen. It was therefore possible to periodically measure the thermal voltage by shutting the current supply off. Plots of the output voltage versus thermal voltage (Vth) were generated for each test specimen and are shown in figures 25 through 27. The data were fitted with a linear equation using a least squares regression to yield thermal voltage as a function of output voltage reading. This thermal voltage was calculated at each data point and subtracted from the output voltage reading.

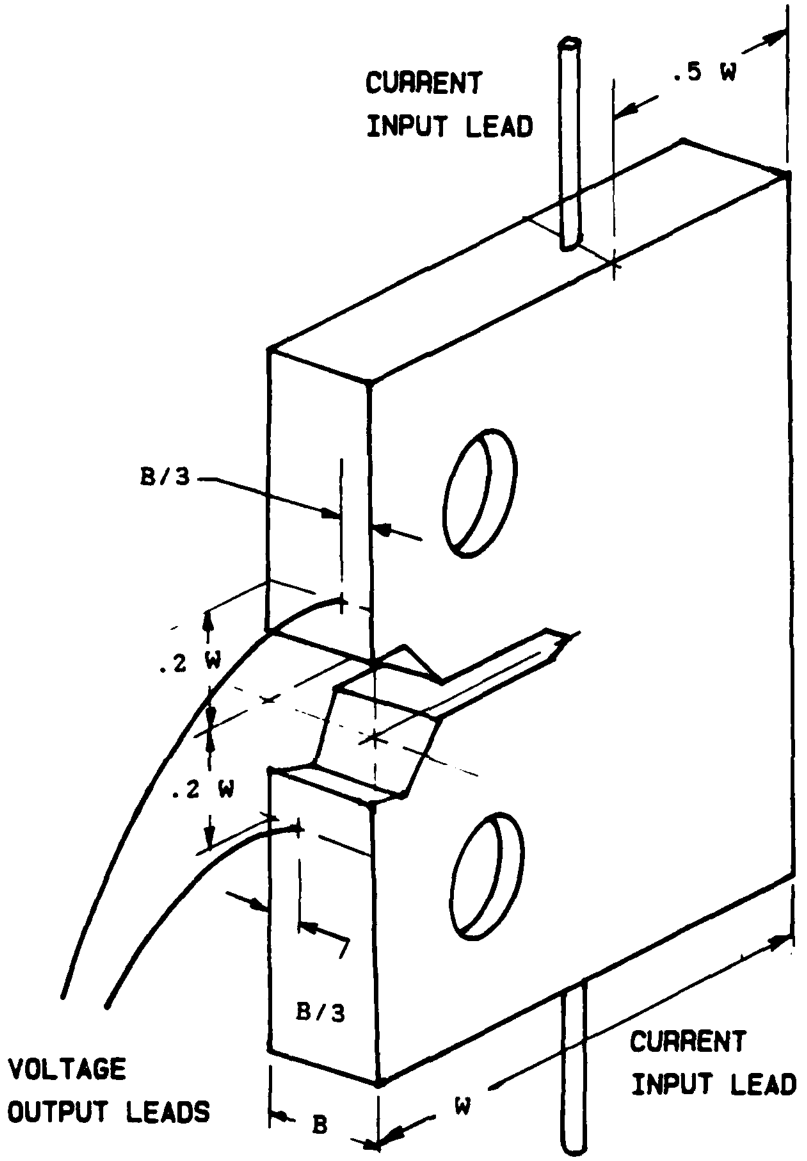


Figure 24 Location of Input and Output Leads on Specimen.

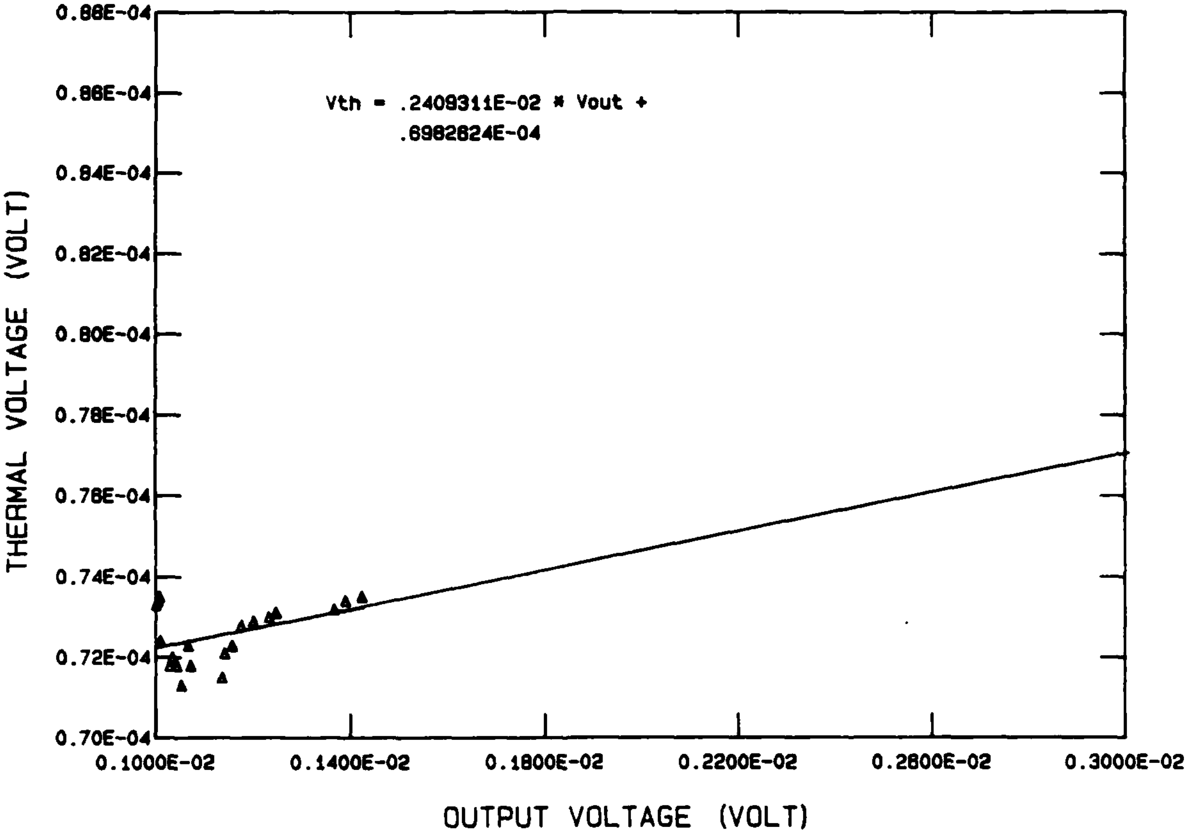


Figure 25 Thermal versus Output Voltage

Specimen 84-502.

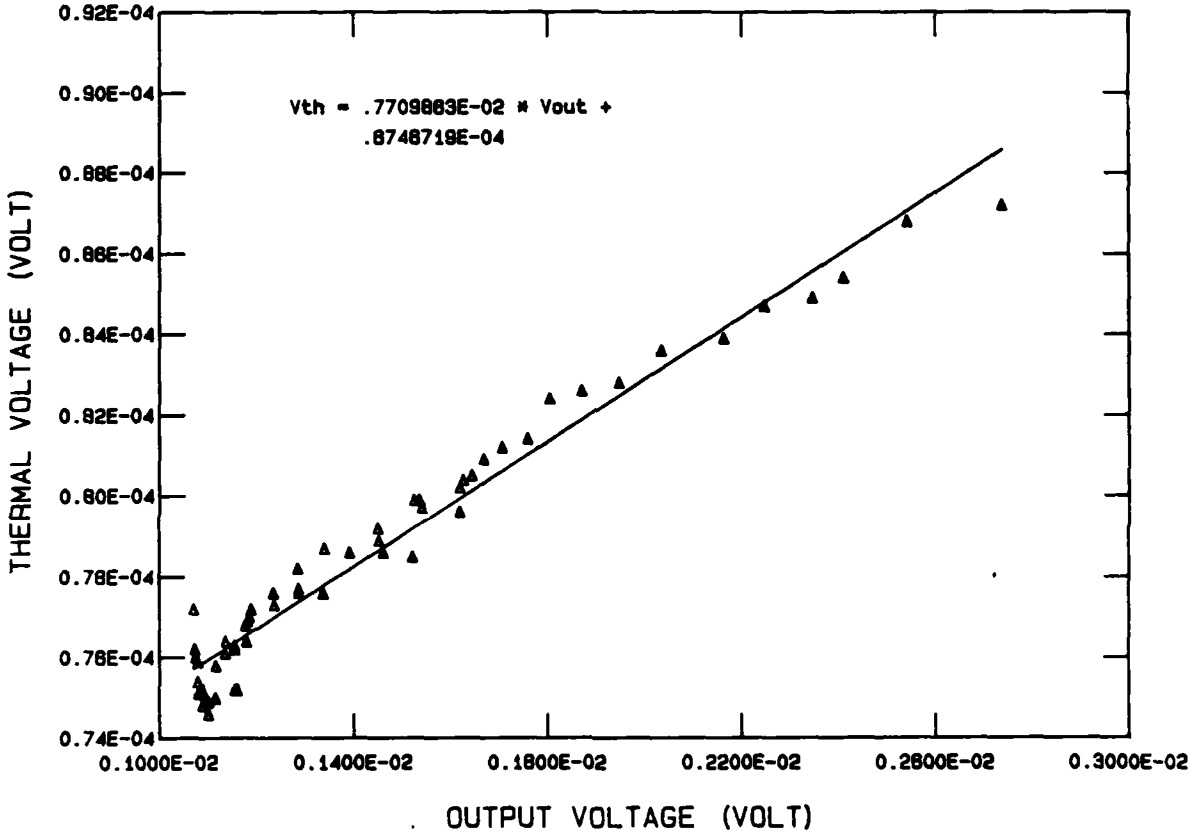


Figure 26 Thermal versus Output Voltage

Specimen 84-503.

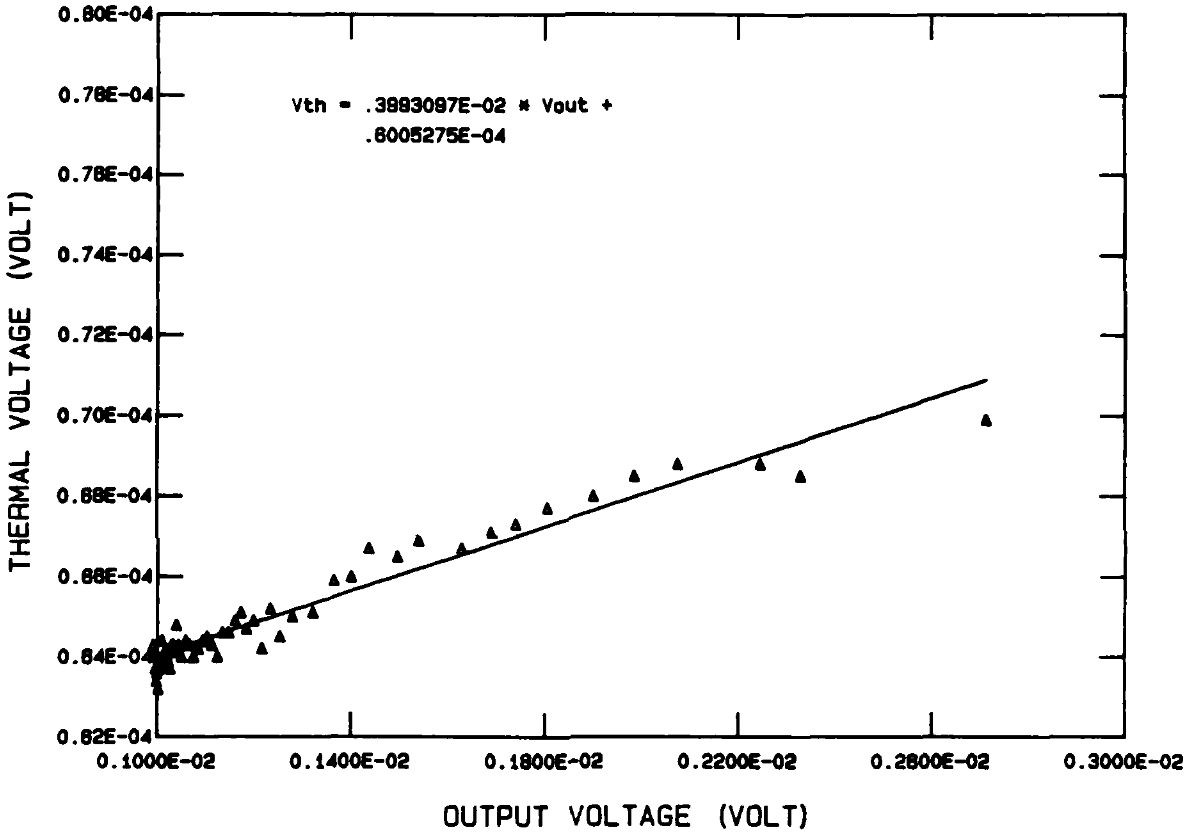
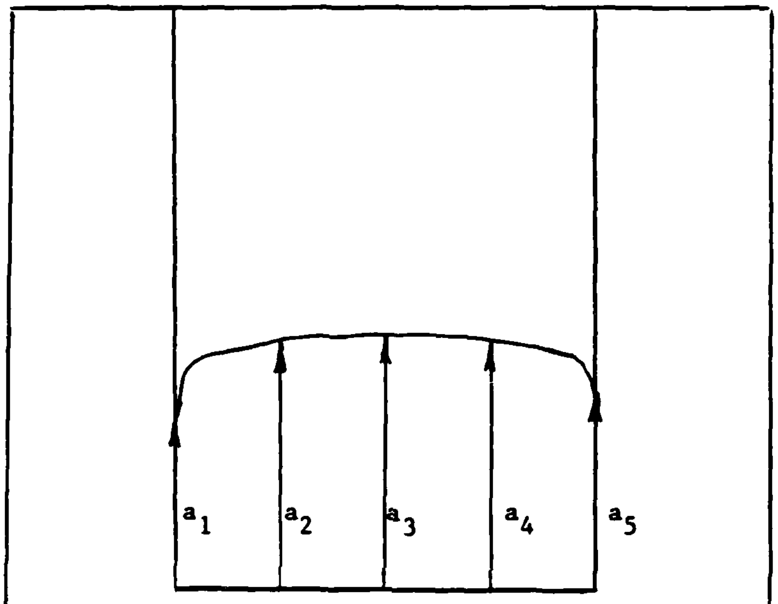


Figure 27 Thermal versus Output Voltage

Specimen 84-504.

Specimen 84-504-EE5 was tested to provide a calibration curve relating crack length to output voltage under conditions of sustained-load crack growth with no applied overloads. The initial voltage reading, minus thermal voltage, was called the initial reference voltage Vo. The initial crack size corresponding to Vo was measured using a method similar to the one recommended in ASTM E399 [16]. An optical microscope, equipped with a measurement table, was used to make a five point through-the-thickness measurement of the initial crack tip profile on the ruptured specimen. The weighted average, defined in figure 28, was used to calculate an average crack size and a crack tunneling correction factor to account for the increased crack depth at the center of the crack front. The tunneling correction factor was used to bias the surface optical measurements, made at locations a_1 and a_5 in figure 28, to obtain a average crack depth. Calibration specimen 84-504-EE5 had an average tunneling value of 0.445 mm. This value was added to each optical reading to get a corrected a_{opt} value.

A functional relationship between a_{opt} and voltage readings (minus thermal voltage, and normalized to the referenced voltage Vo) was developed using a least squares polynomial fitting program. H. H. Johnson [17] developed complex functions for calibrating crack length measurements to voltage. However, Johnson's results showed that the crack-starter geometry strongly influences the



$$a_{opt} = (a_1+a_5)/2$$

$$a_{avg} = ((a_1)/2+a_2+a_3+a_4+(a_5)/2)/4$$

$$a_t = a_{avg}-a_{opt}$$

Figure 28 Average Crack Size and Tunneling Correction Term.

calibration between crack length and measured potential. Therefore for simplicity, an eighth order polynomial was used which was found to adequately fit the experimental data and is shown in figure 29. The polynomial equation for crack length as a function of normalized voltage is,

$$\begin{aligned} a = -332.1731 &+ 1503.861\ (V/Vo)^{1} - 2932.647\ (V/Vo)^{2} \\ &+ 3220.104\ (V/Vo)^{3} - 2176.264\ (V/Vo)^{4} \\ &+ 927.0209\ (V/Vo)^{5} - 243.1114\ (V/Vo)^{6} \\ &+ 35.90090\ (V/Vo)^{7} - 2.286718\ (V/Vo)^{8} \quad \text{(inch)} \end{aligned} \tag{28}$$

The inverse of this function, normalized voltage as a function of a_{opt}, was also generated and is shown in figure 30. The corresponding polynomial equation is,

$$\begin{aligned} V/Vo = -29.66971 &+ 364.3393\ (a)^{1} - 1859.148\ (a)^{2} \\ &+ 5319.197\ (a)^{3} - 9330.943\ (a)^{4} \\ &+ 10274.08\ (a)^{5} - 6926.514\ (a)^{6} \\ &+ 2612.860\ (a)^{7} - 422.2600\ (a)^{8} \end{aligned} \tag{29}$$

where (a) is measured in inches.

This equation was used to calculate the initial voltage (Vo) for specimens with different initial crack sizes. Knowing the initial crack size and corresponding voltage reading (V), an effective Vo was calculated using equation 29. Substituting the ratio of V/Vo_{eff} into equation 28 yields the correct initial flaw size for the specimen.

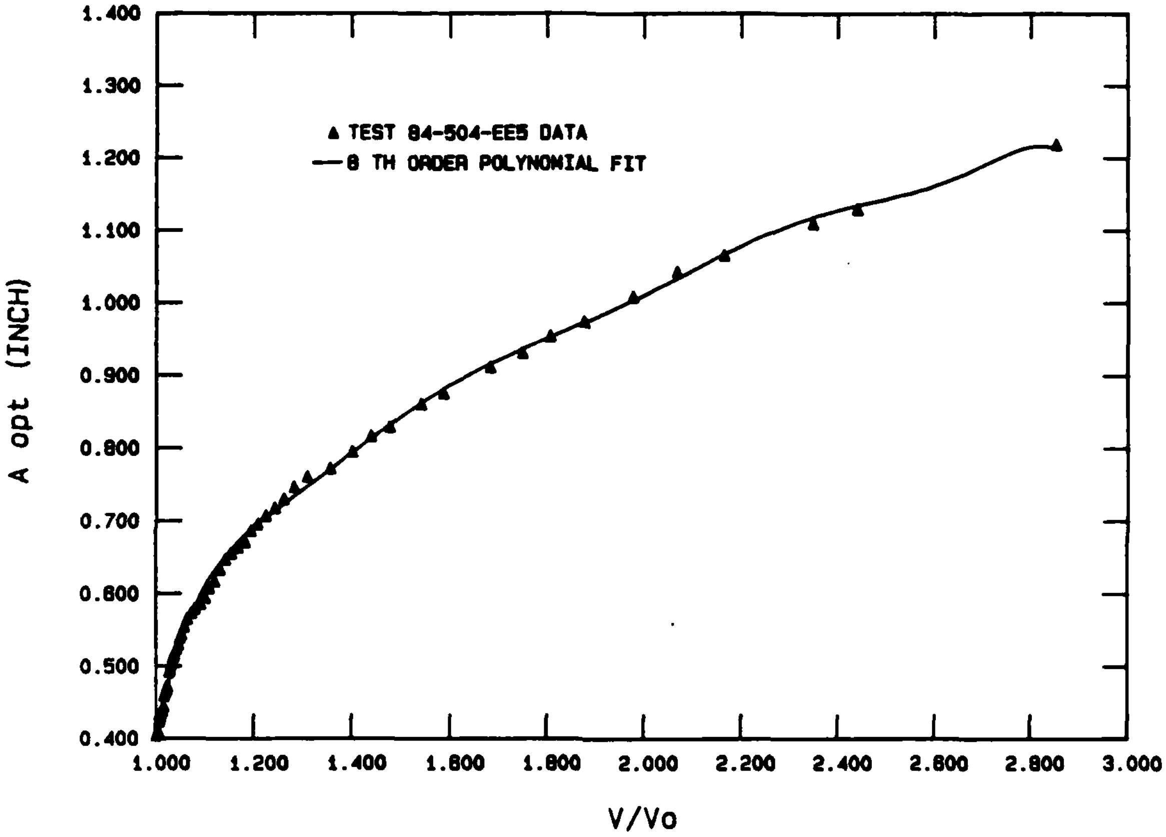


Figure 29 Calibration Curve A_{opt} versus V/V_o.

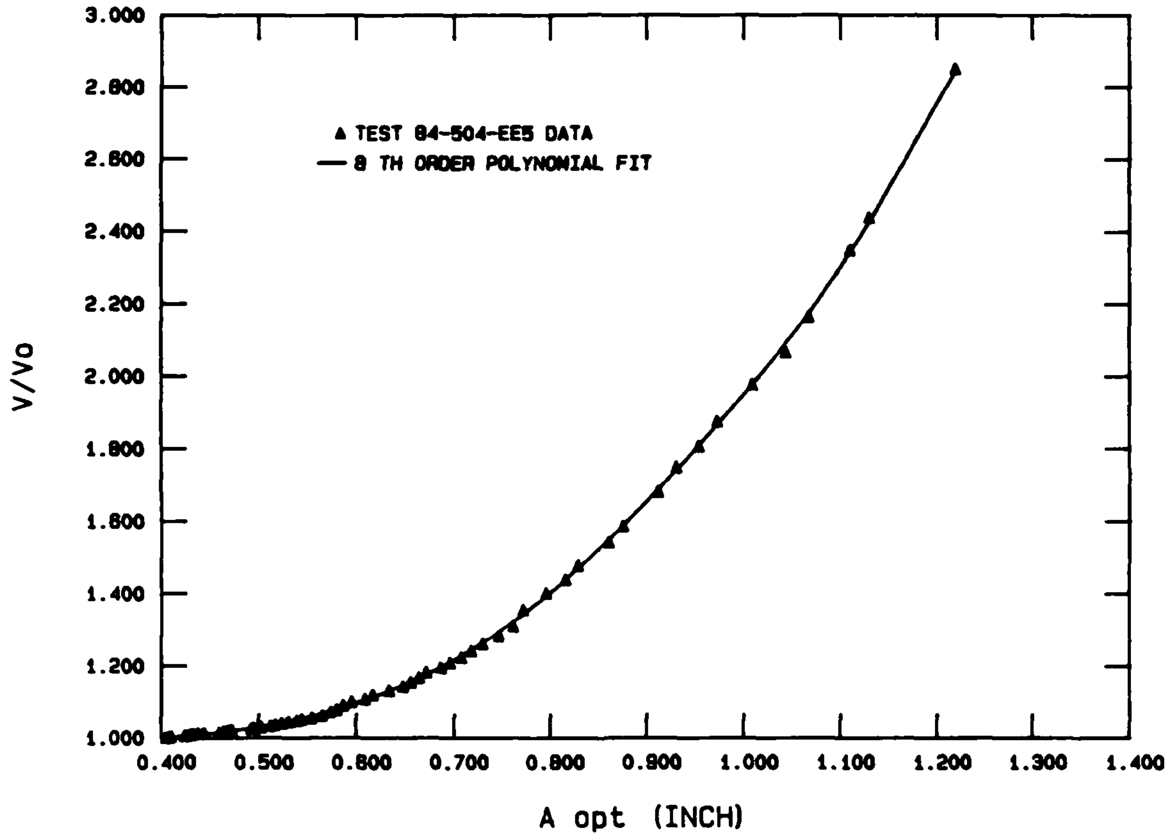


Figure 30 Calibration Curve V/V_o versus A_{opt}.

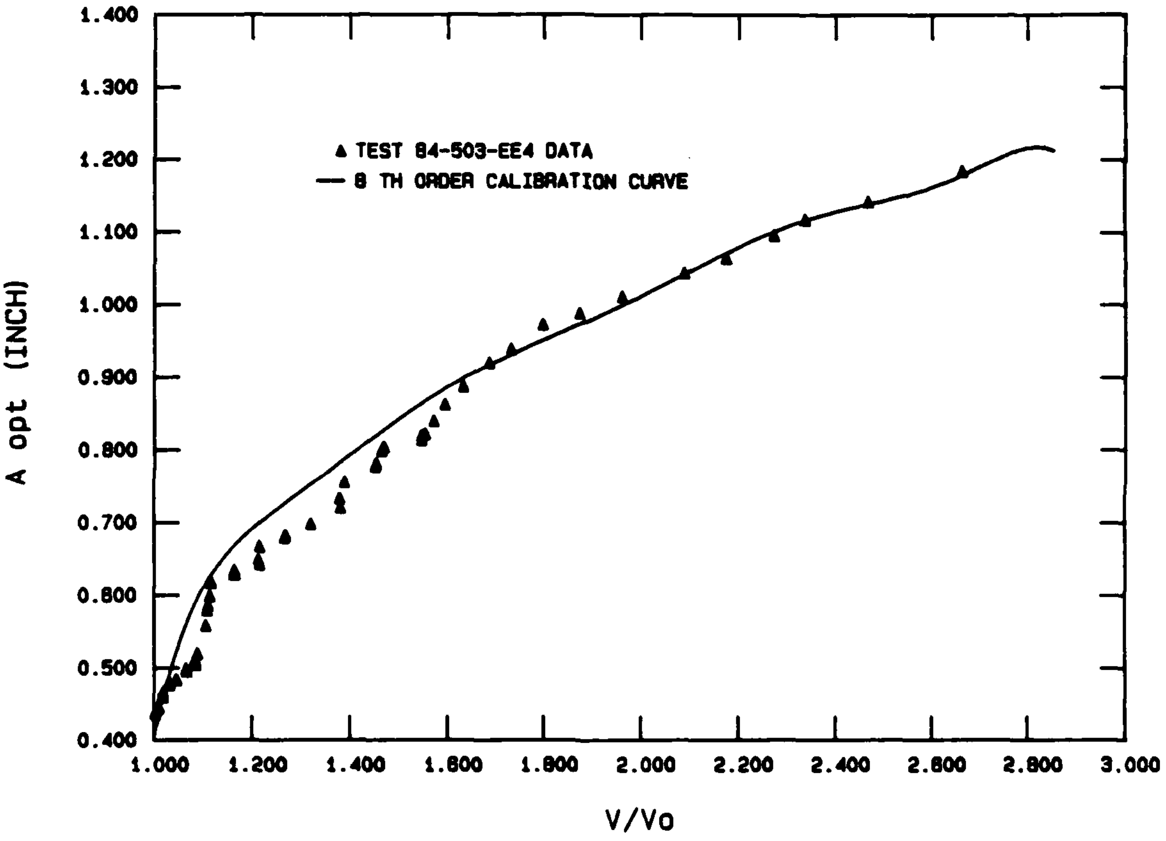


Figure 31 Overload Effect on Calibration Data.

In the tests involving overloads, it was found that each overload caused the a_{opt} versus voltage to shift from the calibration curve. This effect is shown in figure 31. After each overload, there is a period where voltage increases without an appreciable increase in the optical crack size. One explanation is that the crack changes shape with the edge breaking through during the overload cycle. This would account for the jump in optical measurement after each overload. As the crack starts to grow again, evidenced by increasing voltage readings, the surface growth may lag since the crack may tunnel to restore its preferred flaw shape. Although the exact mechanism that cause the deviations from the calibration curve are not fully understood, their effect may be negated by recalibrating the equation after each overload. The procedure is the same one used to calibrate the equation to different initial flaw sizes. Using equation 29 and the a_{opt} and voltage reading taken after each overload, a new Vo was calculated. This Vo was used to normalize voltage readings until the next overload.

The experimental voltage readings recorded by the Tektronix 4051 computer were reduced using the procedures described in this section. The resulting crack growth histories were compared with the analytical predictions and are discussed in the next section.

VI. Experimental Results and Discussion

Experiments were performed in the course of this study in order to obtain additional crack growth data for evaluating the prediction capability of the retardation models. An electric potential crack-measurement system was used for taking crack length measurements during sustained loading with periodic overloads. The applicability of this system to crack growth following overload was investigated.

Three specimens were used in the experimental work. One was used as a calibration specimen for the electric potential system. The other two specimens were used for the proof tests. In addition, a third proof test, conducted by Harms, was also used to evaluate the retardation models.

The crack growth prediction for the calibration specimen 84-504 is shown in figure 32. This specimen was subjected to sustained loading at 650 C with no overloads. As expected, both the Overload and Wheeler models predicted the same growth, since no retardation cycles were applied. The analytical prediction for time to failure was within 20% of the actual test data. This is well within the normal 2X scatter in crack growth data associated with variations in material properties.

Specimen 84-503 was tested with 20 % overloads applied each hour. The test data and retardation model predictions are shown in figure 33. The test was divided into two

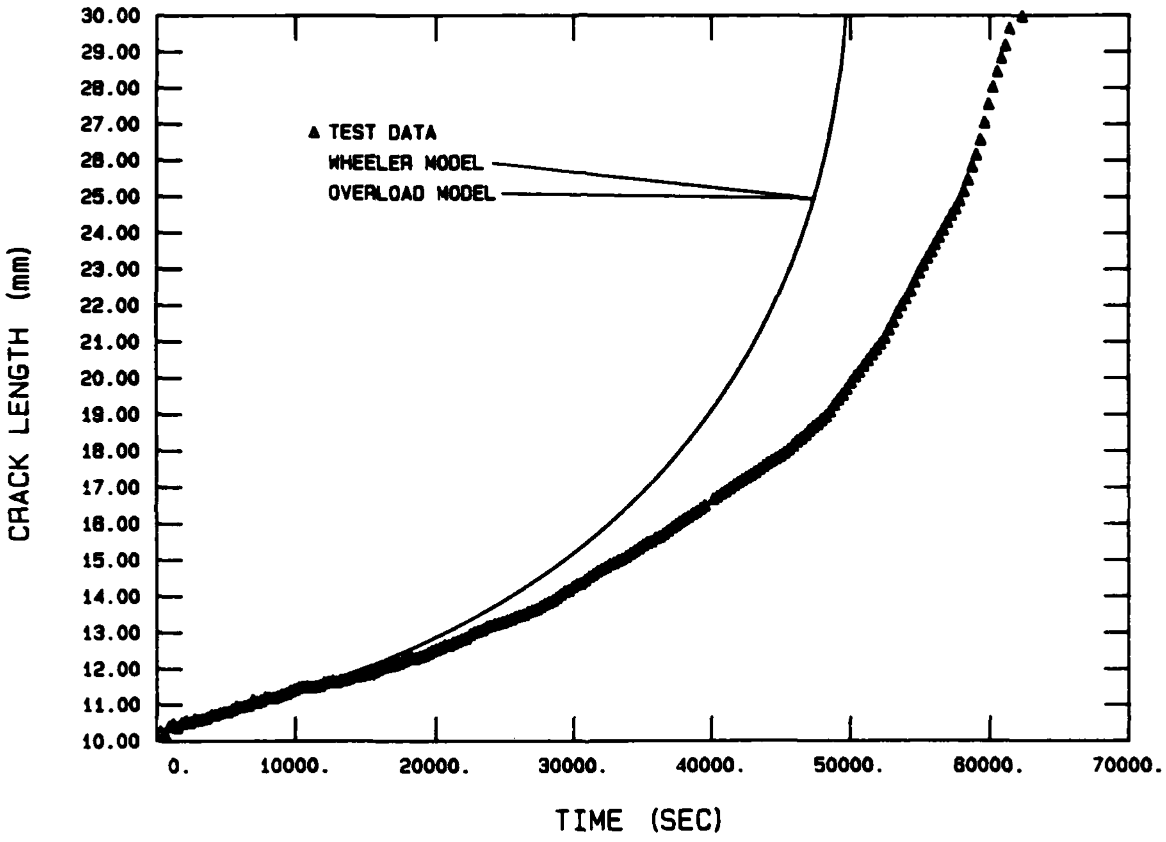


Figure 32 Crack Length versus Time Specimen 84-504.

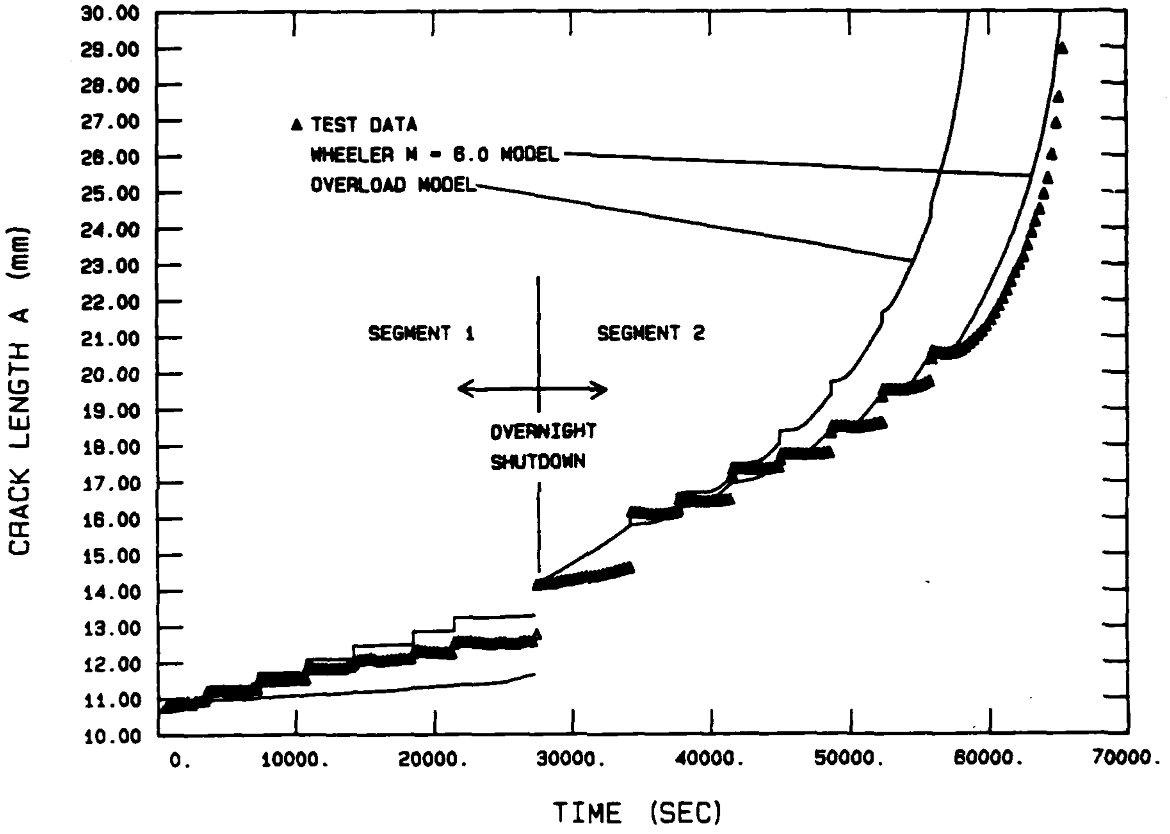


Figure 33 Crack Length versus Time Specimen 84-503.

segments, due to an overnight shutdown in the testing. In the first segment, the crack growth appears to be dominated by the crack jump due to the overload cycle. The Overload model overestimated this jump, while the calculated growth increment from the 0.01 Hz overload cycle before the Wheeler model is applied underestimated it. During the overnight shutdown, the load was removed from the specimen while the temperature was held constant at 650 C. Since the specimen experienced thermal soaking during this shutdown, reloading of the specimen was followed by at least an hour of sustained load growth to re-establish the previous growth rate. The overnight shutdown provided a heat tinted region visible on the fracture surface after the specimen fractured. Optical readings were taken of the crack tip profile to adjust the tunneling correction factor which was then added to the optical surface measurements. The Wheeler model overestimated the total time to failure for the second segment by 1 %. The Overload model underestimated the growth in the same segment by approximately 16 % .

Specimen 84-502 was tested with 20 % overload cycles applied at low 30 (MPa $m^{1/2}$), medium (40 MPa $m^{1/2}$) and high 50 (MPa $m^{1/2}$) stress intensity levels. The effects of the jumps on the crack length, due to overload cycles, was minimized by applying just three overload cycles during the test. The test data and retardation predictions are shown in figure 34. Both models predicted the total time of

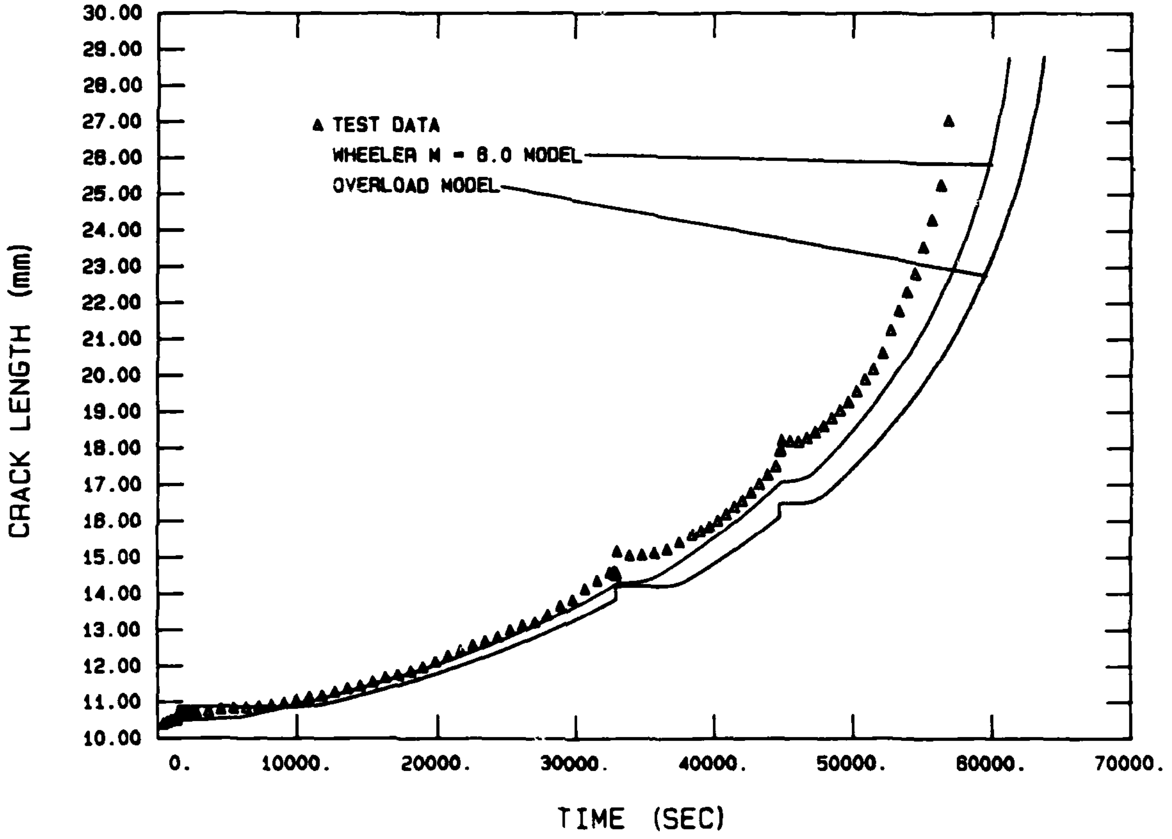


Figure 34 Crack Length versus Time Specimen 84-502.

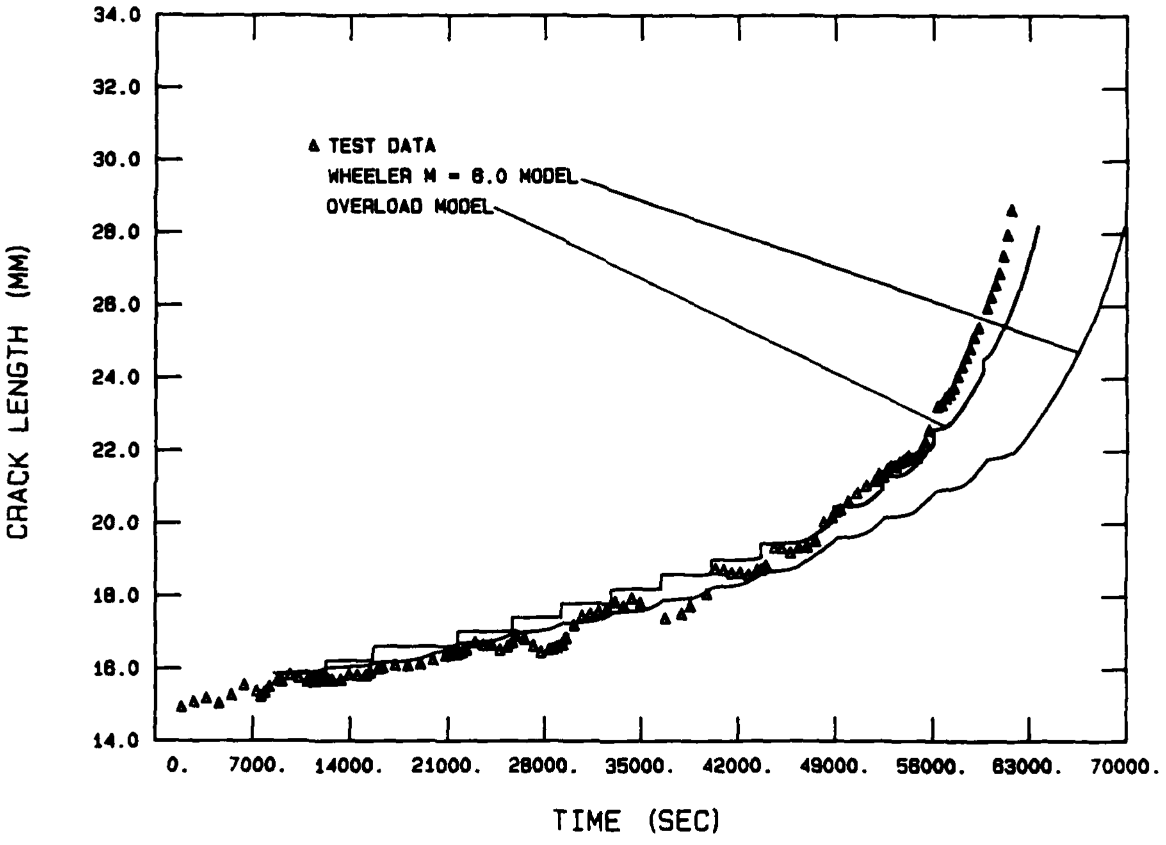


Figure 35 Crack Length versus Time Specimen 84-507.

growth within 10 % of the test data. It was also noted that the retardation predictions were similar in shape to the test data. In general, the models gave better predictions of total time-to-failure when the effect of the crack jumps was minimized.

In addition to the tests conducted as part of this study, the proof test conducted by Harms was analyzed. The test data and retardation predictions are shown in figure 35. The test spectrum consisted of 20 % overload applied each hour. The prediction of the Overload model was within 4 % of the total time-to-failure, but was dependent on the jump in crack length to account for crack growth at lower stress intensity levels. The Wheeler model again underestimated the growth due to the overload cycles and therefore predicted a growth time longer than the actual data.

Accurate prediction of the retardation caused by overload cycles requires empirically defining the α parameter for the Overload model and the shaping exponent m for the Wheeler model. These variables were defined for a specific overload ratio, temperature and test geometry. Both α and m were found to depend upon the stress intensity level at overload application. Using the predefined values of α and m from section IV, both models predicted the retardation effect of overload cycles, when the cycles were applied far enough apart so that no interations occured between cycles. When the overload cycles were applied at closer intervals, the crack growth was dominated by the jumps in the crack length caused by the overload cycles. As seen in the predictions in figures 33 and 35, the Overload model did not accurately in predict the jump in crack length when overload cycles were applied. Defining a function relating the jump in crack length to the current stress intensity level was difficult due to the large scatter in crack length jump measurements. An attempt was made to reduce the scatter in crack length jump measurements by using an electric potential system to measure the crack length. This attempt was unsuccessful mainly due to the electric potential system requiring recalibration via an optical crack measurement after each overload. Optical crack measurements on a specimen under sustained loading are difficult because the crack tip is not sharply defined. Instead, the crack tip looks like a deformation zone, with the exact tip location unknown. The accuracy of both models for predicting crack growth when the overload cycles interact could be improved by further refinement of the jump function.

Overall both retardation programs predicted the total time to failure within the normal 2X scatter associated with variations in material properties. Using the CRACKS program it is possible to analyze spectra of engine fatigue cycles with the sustained-load growth between cycles included.

VII. Conclusions and Recommendations

Conclusions

During the course of this study several observations were make on the applicability of using existing retardation models, developed for airframe application, to predict sustained load crack growth retardation. It was found that:

1.) The crack growth rate for sustained loading (da/dt) and overload fatigue cycles (da/dn) can be represented by one crack growth rate equation. This is accomplished by modeling the sustained load time as equivalent fatigue cycles. The equivalent cycle's period and R ratio are adjusted to obtain the overload fatigue cycle's crack growth rate. The equivalent sustained load fatigue cycles and overload cycles were analyzed using the CRACKS crack growth prediction program developed for airframe cyclic loading. The CRACKS program is capable of predicting the total time to failure within 20 % of experimental data.

2.) The Overload and Wheeler retardation models depend on empirical parameters that are related to the stress intensity level at overload application.

3.) The jump function dominates the crack growth in the Overload model when the overload cycles are spaced close enough to interact.

4.) The Willenborg retardation model modifies the fatigue cycle's R ratio when accounting for retardation.

Thus, the retardation model was dependent on the R ratio chosen for the equivalent sustained load fatigue cycles and was deemed unacceptable for use.

5.) The electric potential crack measurement system is affected by overload cycles and requires recalibration after an overload cycle is applied.

This study verified that the Overload retardation model predicts sustained-load crack growth with periodic overload within normal test data scatter. In addition, procedures were developed to convert sustained loading into equivalent fatigue cycles and analyze crack growth using the CRACKS program. The CRACKS program, with the Wheeler retardation model selected to account for retardation effects, is capable of predicting the total time to failure within 20 % for tests consisting of sustained-load with periodic overloads.

The modified CRACKS program offers the unique capability to analyze sustained-loading and fatigue cycle loading together.. This capability can be readily applied to complex engine spectra, consisting of fatigue cycles with hold times, to predict the crack growth in engine components.

Recommendations

In this study it was found that the electric potential system for measuring crack length was affected by periodic overloads. Additional investigation is required to understand how electric potential crack measurements made on fatigue loaded specimens with periodic overloads compare with sustained-loaded specimens with periodic overloads.

This study used Inconel 718 material exclusively for experimental testing. Additional testing should be performed to determine if the models apply to other materials. Another nickel-base superalloy such as Rene 95 is recommended.

Finally, the retardation parameters for the Overload and Wheeler models should be developed for a wider range of overload ratios and used to analyze a more complex spectrum.

Bibliography

1. Hill, R. J., Reimann, W. H., and Ogg, J. S., "A Retirement-For-Cause Study of an Engine Turbine Disk:, AFWAL-TR-81-2094. Wright-Patterson AFB, OH 1981.

2. Harris, J. A., Sims, D. L., and Annis, C. G., "Concept Definition: Retirement for Cause of F100 Rotor Components", AFWAL-TR-80-4118. Wright-Patterson AFB. OH 1980.

3. Military Standard: Engine Structural Integrity Program, MIL-STD-1783 (USAF) 30 November 1984.)

4. Engle, R.M., "CRACKS, A FORTRAN IV Digital Computer Program for Crack Propagation Analysis," AFFDL-TR-70-107, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Oh, Oct. 1970

5. Wheeler, O.E., "Spectrum Loading and Crack Growth", Journal of Basic Engineering, March 1972, pp 181-186.

6. Willenborg, J., Engle, R.M., and Wood, H.A., "A Crack Growth Retardation Model Using an Effective Stress Concept", AFFDL-TM-71-1-FBR, Wright-Patterson AFB, Oh, Jan 1971

7. Harms K. E., "Overload Effects on Sustained Load Crack Growth at Elevated Temperatures", AFIT Thesis, AFIT/GAE/AA/84D-8, Wright-Patterson AFB, Oh, 1984.

8. Weerasooriya, T., Nicholas, T., "Overload Effects in Sustained Load Crack Growth in Inconel 718", AFWAL-TR-85-4121. Wright-Patterson AFB, OH 1985.

9. Annual Book of ASTM Standards, Part 10; Metals-Mechanical, Fracture, and Corrosion Testing; Erosion and Wear; Effect of Temperature, E 647-83, page 719, 1983.

10. Utah, D. A. General Electric Company, "Crack Growth Modeling in an Advanced Powder Metallurgy Alloy", AFWAL-TR-80-4098, Wright-Patterson AFB, OH 1980.

11. Kreyszig, E., Advanced Engineering Mathematics, Fifth Edition, New York, John Wiley and Sons, 1983, pp 788-789.

12. Larsen J. M. and Nicholas, T. "Load Sequence Crack Growth Transients in a Superalloy at Elevated Temperature", Fracture Mechanics: Fourteenth Symposium Volume II: Testing and Applications, ASTM Society for Testing and Materials, 1983, pp II-536-II-552.

13. Haritos, G.K., Nicholas, T. and Painter, G.O., "Evaluation of Crack Growth Models for Elevated Temperature Fatigue," Eighteenth National Symposium on Fracture Mechanics, ASTM STP XXX, American Society for Testing and Materials, 1985, pp xx-yy

14. Engle, R. M. "Damage Accumulation Techniques in Damage Tolerance Analysis", ASTM STP 842, American Society for Testing and Materials, 1984, pp. 25-35.

15. Collins, J. A., Failure of Materials in Mechanical Design, New York, John Wiley and Sons, 1981, Ch. 8.

16. Annual Book of ASTM Standards, Part 10; Metals-Mechanical, Fracture, and Corrosion Testing; Erosion and Wear; Effect of Temperature, 1981.

17. Johnson, H. H., "Calibrating the Electric Potential Method for Studying Slow Crack Growth", Materials Research and Standards, Vol. 5, No. 9, 1965

Appendix A

Heat Treatment History of Test Specimens

Anneal at 968 C for 1 hour - air cool

Age harden at 718 C for 8 hours - furnace cool to 621 C

Age harden at 621 C for a total of an additional 10 hours

Appendix 8.

OVERLOAD PROGRAM

```
C     PROGRAM OVERLD
      DOUBLE PRECISION V
      REAL*4 LOWER,KUP,KLOW,N,LOWW,KINIT,LOWERR,KLOWW,
     1KUPP,LOWR,LOAD,K
      INTEGER*4 PIECES,D,YNCR,NAME(7)
      DIMENSION TITLE(60),SBTITL(60),TIME(20),OL(20)
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
C     DEFAULTS FOR DIAGNOSTIC OPTION
      DATA V/0/,PI/3.14159/,TOL/.0001/,R/.0/,ZERO/0.0/,T/.25/,TRIG/1./
C ***** OPEN FILES    2:INPUT   1:OUTPUT   3:PLOT   ************************
      WRITE (5,40)
   40 FORMAT (' WHAT IS YOUR INPUT FILE? ',$)
      READ (5,90) NAME
      OPEN (UNIT=2,NAME=NAME,TYPE='OLD')
      WRITE (5,50)
   50 FORMAT (/,' WHAT SHOULD YOUR OUTPUT FILE BE CALLED? ',$)
      READ (5,90) NAME
      OPEN (UNIT=1,NAME=NAME,TYPE='NEW')
      WRITE (5,60)
   60 FORMAT (/,' WHAT SHOULD YOUR PLOT FILE BE CALLED? ',$)
      READ (5,90) NAME
      OPEN (UNIT=3,NAME=NAME,TYPE='NEW')
   90 FORMAT (11A2)
C ***** ALL FILES ATTACHED ****************************************
C
C ***** READ INPUT FILE *******************************************
      READ (2,1120) TITLE
      READ (2,1120,END=1590) SBTITL
      READ (2,*) TOUGH,LOWW,R,YIELD
      READ (2,*) BEE,DKSTAR,PEA,QUE,DEE
      READ (2,*) B,LOAD,W
      WRITE (5,105)
      WRITE (5,110)
  105 FORMAT (/,' DO YOU WANT TO BREAK OUT ON A CRACK LENGTH')
  110 FORMAT (' OR AT THE KIC STRESS INTENSITY [1=A, 0=KIC]-->',$)
      READ (5,*) AFIN
      IF (AFIN.EQ.1) GO TO 150
      AFIN=10000.
      GO TO 160
  150 WRITE (5,155)
  155 FORMAT (/,' INPUT YOUR FINAL CRACK LENGTH, INCHES-->',$)
      READ (5,*) AFIN
C ***** END OF INPUT ***********************************************
C
C ***** PRINT OUTPUT FORMATS ON FILES ***************************
```

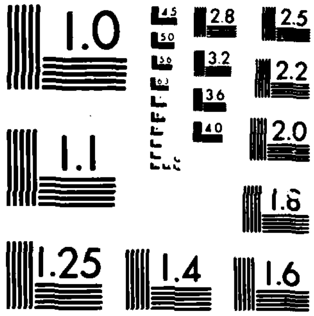
```
 160  WRITE (1,1130) TITLE
      WRITE (3,1130) TITLE
      WRITE (5,1130) TITLE
      WRITE (1,1130) SBTITL
      WRITE (3,1130) SBTITL
      WRITE (5,1130) SBTITL
      WRITE (1,740) TOUGH,R
      WRITE (1,850) YIELD
      WRITE (1,830) B,W
      WRITE (1,930) BEE,DKSTAR,PEA
      WRITE (1,940) DKSTAR,QUE
      WRITE (1,950) TOUGH,DEE
      WRITE (1,1270)
      WRITE (1,1280)
 740  FORMAT (' FRACTURE TOUGHNESS:',F6.2,/,' STRESS RATIO:',F5.3)
 830  FORMAT (' THICKNESS: ',F7.5,/,' WIDTH:',F7.5)
 850  FORMAT (' YIELD STRENGTH:',F6.2)
 930  FORMAT (' da/dN=EXP[',F6.2,']*[[DELTA-K/',F6.2,']**',F6.2,']*')
 940  FORMAT (' [[ln[DELTA-K/',F6.2,']]**',F6.2,']*')
 950  FORMAT (' [[ln[',F6.2,'/DELTA-K]]**',F6.2,']')
1120  FORMAT (60A2)
1130  FORMAT (1X,60A2)
1270  FORMAT (35X,' delta')
1280  FORMAT (1X,' N(x1000)',10X,'a',9X,'LOAD',2X,
     1'  K  ',1X,'  da/dN  ',1X,'PIECES')
C ***** END OF HEADERS ***************************************************
C
C ***** READ IN SPECTRUM LOAD *************************************
      NA=1
  95  READ (2,*,END=100) TIME(NA),OL(NA)
      NA=NA+1
      GO TO 95
 100  NA=NA-1
C *****************************************************************
      TOUGH=(1.0-R)*TOUGH
      DKCRIT=TOUGH
      TOUGH=TOUGH*.95
1210  STAR=1.0
      KUP=0
      V=TIME(1)
      SIGMA=LOAD
C ***** START OF SPECTRUM INTEGRATION *****************************
      DO 1570 NS=1,NA
      DAGER=OL(NS)
      NSKIP=0
      DOL=0
      U=LOWW
1290  LOWER=U
C ******************************************************************
C
1320  CALL SIMP(SUM,LOWER,UPPER,SIGMA,KLOW,KUP,PIECES,YIELD,STAR,PYTCH,
     1CALPHA,BETA,DOL)
```

```
C
C ***** Check if initial K is greater than K1c *********************
      IF (KLOW.LT.TOUGH)GO TO 1360
      WRITE (5,1330)
      WRITE (1,1330)
1330  FORMAT (10X,' *****SPECIMEN FAILED ON LOADING*****')
      GO TO 1490
C
1360  VN=V
      V=V+(SUM/1000)
C ***** CHECK IF INTEGRATED TIME IS MORE THAN THE NEXT OVERLOAD *
C
      IF (NS.EQ.NA) GO TO 1370
      IF (V.LT.TIME(NS+1)) GO TO 1370
      IF (NSKIP.EQ.1) GO TO 1370
      PYTIT=PYTCH*(TIME(NS+1)-VN)/(SUM/1000.)
      UPPER=LOWER+PYTIT
C ************ EXTRA PRINT TO CHECK LINEAR PYTCH INCREMENT *****
C      WRITE (1,1331) V,VN,NS,PYTIT,LOWER,UPPER,SUM
C1331  FORMAT (1X,'V= ',F15.6,' VN = ',F15.6,'  NS= ',I2,/,'  PYTIT= ',
C     1F15.10,/,' LOWER= ',F16.13,'UPPER= ',F16.13,' SUM = ',F9.3)
C **************************************************************
      V=VN
      NSKIP=1
      GO TO 1320
C
1370  CALL FINDF2 (KLOW,F2)
      DADNL=1/F2
      CALL FINDF2 (KUP,F2)
      DADNU=1/F2
      IF (U.EQ.LOWW) WRITE (1,1421)VN,LOWER,SIGMA,KLOW,DADNL,PIECES
C
C *****  CONVERSION FROM ENGLISH TO METRIC TO WRITE PLOT FILE *******
C
      CBETA=BETA
      CBETA=CBETA/25.4
C     1/INCHES-->1/MILLIMETERS
C
      LOWERR=LOWER*25.4
C     INCHES-->MILLIMETERS
      DADNLL=DADNL*0.0254
C     INCHES PER CYCLE-->METERS PER CYCLE
C
      KLOWW=KLOW*1.0989
C     KSI ROOT INCHES-->MPA ROOT METERS
C
      UPPERR=UPPER*25.4
      DADNUU=DADNU*0.0254
      VV=V*1000
      KUPP=KUP*1.0989
      VVN=VN*1000.
C      IF (U.EQ.LOWW) WRITE(3,1375) CALPHA,CBETA
```

```
      IF (U.EQ.LOWW) WRITE (3,*) VVN,LOWERR,DADNLL,KLOWW
      IF (U.EQ.LOWW) WRITE (5,1420)VN,LOWER,SIGMA,KLOW,DADNL,PIECES
      WRITE (1,1420) ,V,UPPER,SIGMA,KUP,DADNU,PIECES
      WRITE (3,*) ,VV,UPPERR,DADNUU,KUPP
      WRITE (5,1420) ,V,UPPER,SIGMA,KUP,DADNU,PIECES
C1375  FORMAT(' ALPHA = ',F8.5,/,' BETA = ',F8.5)
1420  FORMAT (1X,F15.3,F9.6,F9.2,F7.2,E10.3,I7)
1421  FORMAT (1X,F15.4,F9.6,F9.2,F7.2,E10.3,I7)
1430  IF ((KUP.GT.TOUGH).OR.(UPPER.GE.AFIN)) GO TO 1490
      IF (NSKIP.EQ.1)GO TO 1560
      U=U+PYTCH
      CALL CPYTCH (KUP,PYTCH)
      GO TO 1290
1490  WRITE (1,1500)
1500  FORMAT (6X,' C-T SPECIMEN, SANS PLASTIC-ZONE CORRECTION')
      GO TO 1590
C
C ***** ENTER JUMP FUNCTION ************************
C
1560  XJUMP=.015
      LOWW=U+XJUMP
C
      WRITE (1,1561) XJUMP,KUP
1561  FORMAT(1X,'XJUMP = ',F15.10,' KUP= ',F6.2)
1570  CONTINUE
C    ******************** END DO LOOP ***************
1590  CALL EXIT
      END
C     SUBROUTINES IN DESCENDING ORDER OF USE,FIRST,SIMPSON'S RULE APPROX
      SUBROUTINE SIMP(SUM,LOWER,UPPER,SIGMA,KLOW,KUP,PIECES,YIELD,STAR,
     1PYTCH,CALPHA,BETA,DOL)
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
      REAL*4 K,N,LOWER,KLOW,KUP,LOWW
      INTEGER*4 PIECES
      PIECES=2
      X=LOWER/T
      CALL CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      IF (X.EQ.LOWW/T)
     1CALL CAL(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      KLOW=K
      IF (X.EQ.LOWW/T) CALL CPYTCH(K,PYTCH)
      IF (NSKIP.NE.1) UPPER=LOWER+PYTCH
      ESUM=F2
      DELTA=(UPPER-LOWER)/PIECES
      EVSUM=0
      X=(LOWER+DELTA)/T
      CALL CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      ODSUM=F2
      X=UPPER/T
      CALL CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      KUP=K
      ESUM=ESUM+F2
      SUM=(ESUM+4*ODSUM)*DELTA/3
1600  PIECES=PIECES*2
      SUM1=SUM
      DELTA=(UPPER-LOWER)/PIECES
      EVSUM=EVSUM+ODSUM
      ODSUM=0
      L=IFIX(FLOAT(PIECES)/2)
      DO 1610 I=1,10000
      Z=LOWER+DELTA*(2*I-1)
      X=Z/T
      CALL CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      ODSUM=ODSUM+F2
1610  IF (I.EQ.L) GO TO 1620
1620  SUM=(ESUM+4*ODSUM+2*EVSUM)*DELTA/3
      IF (ABS(SUM-SUM1).GT.ABS(TOL*SUM)) GO TO 1600
      RETURN
      END
C     SUBROUTINE FOR CALCULATING CALPHA FOR THE CT RETARDATION
C     MODEL (20% AND 50% OVERLOAD CASES)
      SUBROUTINE CAL(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
      REAL*4 K,N,LOWW
      IF (DAGER.EQ.0.) CALPHA=0.
      IF (DAGER.EQ.0.) GO TO 1679
      SK=K/DKSTAR
      CAS=1-1/SK
      IF (DAGER.EQ.50) GO TO 1675
      BETA=(SQRT(2.0)*PI**2*YIELD**2)/(.44*K**2)
      CACAS=(-.730791E-01)*SK**3 + (.303086)*SK**2 -
     1 (.422108)*SK + 0.117517E01
      GO TO 1677
1675  BETA=(SQRT(2.0)*PI**2*YIELD**2)/(1.25*K**2)
      CACAS=(-.121127E-01)*SK**2 + (.239231E-01)*SK + 0.987133
1677  CALPHA=CACAS*CAS
1679  DOL=1
      CALL CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      RETURN
      END
C     SUBROUTINE FOR COMPACT TENSION SPECIMENS
C     IN THIS ROUTINE, "SIGMA" IS A LOAD, NOT A STRESS!!!
      SUBROUTINE CT(SIGMA,X,K,F2,YIELD,STAR,PYTCH,CALPHA,BETA,DOL)
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
      REAL*4 K,M1,M2,M3,N,LOWW
      ALPHA=X*T/W
      M1=SIGMA/(B*(SQRT(W)))
      M2=(2+ALPHA)/((1-ALPHA)**1.5)
```

AD-A164 018 PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD 2/2
CRACK GROWTH IN A H..(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. R L HASTIE
UNCLASSIFIED DEC 85 AFIT/GA/AA/85D-6 F/G 11/6 NL

END
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963 A

```
      M3=0.886+(4.64*ALPHA)-(13.32*(ALPHA**2))+
     1(14.72*(ALPHA**3))-(5.6*(ALPHA**4))
      K=M1*M2*M3*(1.0-R)
      IF ((X.EQ.LOWW/T) .AND. (DOL.EQ.0)) GO TO 1760
      DELA=X*T-LOWW
      STAR=CALPHA*EXP(-BETA*DELA)
      EFFK=K*(1-STAR)
      K=EFFK
1760  CALL FINDF2(K,F2)
      RETURN
      END
      SUBROUTINE FINDF2(K,F2)
C     SUBROUTINE SELECTING F2 BASED MODIFIED SIGMODAL EQN ONLY
      REAL*4 K,N,LOWW
      INTEGER*4 CINH
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
C ***********************************************
      CINH=2
C ** USE MODIFIED SIGMODAL EQN ONLY CINH=2 *******
      IF (CINH.NE.2) GO TO 1840
      IF (K.GT.DKSTAR) GO TO 1830
      WRITE (5,1800) K
      WRITE (5,1810) DKSTAR
      WRITE (5,1820)
1800  FORMAT (' YOUR INITIAL DELTA-K IS ',F5.2,',AND THIS IS SMALL-')
1810  FORMAT (' ER THAN YOUR DELTA-K THRESHOLD OF ',F5.2,'. THIS')
1820  FORMAT (' WILL GIVE THE MSE INDIGESTION. TRY AGAIN.')
      CALL EXIT
      STOP
1830  F2=1.0/(EXP(BEE)*((K/DKSTAR)**PEA)*((ALOG10(
     1K/DKSTAR))**QUE)*((ALOG10(DKCRIT/K))**DEE))
1840  IF (CINH.EQ.1) F2=1.0/(10.0**(S1*(SINH(S2*((ALOG10(K))+S3)))
     1+S4))
      IF (CINH.EQ.0) F2=(K**(-N))/C
      RETURN
      END
C ***** SUBROUTINE TO CALCULATE PYTCH INCREMENT FOR SIMP *****
      SUBROUTINE CPYTCH(K,PYTCH)
      REAL*4 K
      COMMON/LYIA1/PI,Q,TOL,W,B,T,R,
     1QUE,PEA,BEE,DEE,DKSTAR,TOUGH,DKCRIT,
     2TRIG,LOWW,DAGER,NSKIP
      IF (K .LE. 22.77) PYTCH=.00001
      IF (22.77 .LT. K .AND. K .LE. 30.0)
     1PYTCH=(.002*K-.045)/53.5
      IF (30.0 .LT. K)
     1PYTCH=(.0004*K-.01)/7.0
      RETURN
      END
```

Sample Input

```
TYPE I502T.DAT
30 SEPT 85 SPECIMEN 84-502, EE3 RLH
SPECIMEN  84-502, EE3 AT 1200 DEG F
272.73,.4087,0.,120.
-8.69,21.00,-1.1,1.8,-1.8
.394,2.740,1.5736
0.0,0.0
1.426,20.0
32.888,20.0
44.707,20.0
```

Appendix C.

Modified CRACKS Program

```
      PROGRAM CRACKS4(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE1)
      COMMON/DATA/ EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AMAX,NZERO
      INTEGER EQN
      REAL NZERO
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /              KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /             NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /              AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /              BTABL2(100),ASTART(10),ASTOP(10)
      COMMON/OUTPOT/KMAX,KMAXA,DELTAK,IFLT,DADNPR
      REAL KMAX,KMAXA
      WRITE(6,3000)
      IRSTRT = 0
      MOD1 = 0
      RTARD1 = 0
      ICASE = 0
    1 ICASE = ICASE + 1
      ISPEC=0
    5 IFLT=0
      RETARD = 0
      JSTART=1
      OVLD = 0.0
      IF(RTARD1.EQ.0) GO TO 10
      NORTRD = 0
      MODEL = MOD1
   10 ISPEC = ISPEC + 1
      CALL INPUT(ICASE,ISPEC,IRSTRT)
      IF(ISTOP.NE.0) GO TO 1
      IF(IRSTRT.GE.4) GO TO 1150
   20 A = AZERO
      CYC = NZERO
      ABLK = AZERO
      DO 1000  J1=JSTART,NBLKS
      DO 900 J2=1,NSEGS
      ISEG = ISEGS(J2)
      IBLK = IBLKS(J2)
      NLYR = NLYRS(ISEG)
      DO 800 J3=1,IBLK
      IFLT = IFLT + 1
      DO 700 J4=1,NLYR
      DN = CYCLES(J4,ISEG)
      CYCSVE = CYC
  500 CALL GRWCRK(CYC,A,DN)
      IF(A.LT.AMAX) GO TO 550
      WRITE(6,3600) A,AMAX,IFLT
      GO TO 1100
  550 IF (ISTOP.NE.2) GO TO 600
      DN = CYCSVE + DN - CYC
      ISTOP = 0
      ISURF = 0
      IF(DN.GT.0.0) GO TO 500
  600 IF(ISTOP.NE.0) GO TO 1100
      IF(IFLT.EQ.1) CALL OUTPUT(CYC,A)
      IF(J4PR.EQ.0) GO TO 700
      IF(MOD(IFLT,J4PR).EQ.0) CALL OUTPUT(CYC,A)
  700 CONTINUE
      IF(J3PR.EQ.0) GO TO 800
      IF(MOD(IFLT,J3PR).NE.0) GO TO 800
      WRITE(6,3050) IFLT,ISEG,A
  800 CONTINUE
      IF(J2PR.EQ.0) GO TO 900
      IF(MOD(J2,J2PR).NE.0) GO TO 900
      WRITE(6,3100) J2,J1,A
  900 CONTINUE
      IF(IRSTRT.EQ.3) CALL RESTRT(CYC,A,IRSTRT)
      IF(J1PR.EQ.0) GO TO 1000
      IF(MOD(J1,J1PR).NE.0) GO TO 1000
      DELA = A - ABLK
      GROWTH = A - AZERO
      ABLK = A
      WRITE(6,3200) J1,A,DELA,GROWTH
 1000 CONTINUE
      GROWTH = A - AZERO
      WRITE(6,3500) CYC,A,GROWTH
 1100 IF(NORTRD.EQ.0) GO TO 1300
 1150 ICHECK = IRSTRT + 1
      GO TO (1,5,10,1,1200,1250),ICHECK
 1200 CALL RESTRT(CYC,A,IRSTRT)
      AZERO = A
      NZERO = CYC
      JSTART = J1 - 1
      GO TO 20
 1250 CALL RESTRT(CYC,A,IRSTRT)
      AZERO = A
      NZERO = CYC
      JSTART = J1 + 1
      NBLKS = NBLKS + J1
      GO TO 20
 1300 ISPEC = ISPEC + 1
      WRITE(6,2000)
      WRITE(6,2900) ICASE,ISPEC
      WRITE(6,3300)
      RTARD1 = 1
      MOD1 = MODEL
      MODEL = 0
      ISTOP = 0
      RETARD = 0
      NORTRD = 1
      OVLD = 0.0
      IFLT = 0
      GO TO 20
 2000 FORMAT(1H1)
 2900 FORMAT(1H1,70(1H*)/26X,5HCASE ,I2,4HRUN ,I2/1X,70(1H*))
 3000 FORMAT(1H1, 70(1H$)/1X, 70(1H$)//1X,29(1H*),12H CRACKS IV  29(1H*)
     1// 20X,9HVERSION 6,12X,9H09/21/79 /30X,13HR.M.ENGLE JR./18X,
     2     36HAIR FORCE FLIGHT DYNAMICS LABORATORY//26X,10HAFFDL(FBE)/
     3     26X,19HATTN(R.M.ENGLE JR.)/26X,17HW-PAFB,OHIO 45433//
     4     26X,12H513-255-6104//1X, 70(1H$)/1X, 70(1H$) )
 3050 FORMAT(15H END OF FLIGHT I5,9H  MISSION,I2,7X,14HCRACK LENGTH =,F
     /10.5)
 3100 FORMAT(15H END OF SEGMENT,I6,9H OF BLOCK,I5,5X,14HCRACK LENGTH =,
     /F10.5)
 3200 FORMAT(14H0 END OF BLOCK,I5,20X,14HCRACK LENGTH =,F10.5/
     /5X,19HGROWTH THIS BLOCK =,F10.5,5X,14HTOTAL GROWTH =,F10.5//)
 3300 FORMAT(///1X, 70(1H*)/24X,33HRERUN OF CASE WITH NO RETARDATION/
     /       1X, 70(1H*))
 3500 FORMAT(1H0, 70(1H*)/ 5X,14HTOTAL CYCLES =,F12.2,5X,20HFINAL CRACK
     /LENGTH =,F10.5/22X,20HTOTAL CRACK GROWTH =,F10.5/1X, 70(1H*)/1H1)
 3600 FORMAT(1H0, 70(1H*)/ 1X,20HCURRENT CRACK LENGTHF10.5/31H EXCEEDS A
     /LLOWABLE CRACK LENGTH,F10.5/9X17H IN FLIGHT NUMBER,I6/1X, 70(1H*))
      END
      SUBROUTINE OUTPUT(CYC,A)
C
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /                 NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /                  KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      COMMON/DATA/EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AMAX,NZERO
      REAL KSUBC,KSUBQ
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/OUTPOT/KMAX,KMAXA,DELTAK,IFLT,DADNPR
      REAL KMAX,KMAXA,KMXEFF
      DATA NOP/0/
      IF(RETARD.EQ.0) GO TO 100
C
C     OBTAIN OUTPUT QUANTITIES FOR RETARDATION
C
      GO TO (10,20,30,40,50), MODEL
   10 CALL WHELER(CYC,A,DDNRET)
      GO TO 90
   20 CALL WLNBRG(CYC,A,DDNRET)
      GO TO 90
   30 DDNRET = DADNPR
      GO TO 90
   40 CONTINUE
   50 CONTINUE
   90 KMXEFF = KMAX
      DLKEFF = DELTAK
C
C     DETERMINE UNRETARDED QUANTITIES CORRESPONDING TO CURRENT CRACK
C     LENGTH
C
  100 CONTINUE
      SIGMAX = SMAX(J4,ISEG)
      SIGMIN = SMIN(J4,ISEG)
      CALL RATE(CYC,A,DADN)
      IF(MODEL.NE.0.AND.RETARD.NE.0) GO TO 200
      KMXEFF = KMAX
      DLKEFF = DELTAK
      DDNRET = DADN
C
C     PRINT HEADER AT TOP OF EACH PAGE
C
  200 CONTINUE
      IF(J4.EQ.1) WRITE(6,2000)
      IF(J4.EQ.1 .AND. NOP.EQ.0) WRITE(1,2000)
      IF(J4.EQ.1 .AND. NOP.EQ.0) WRITE(1,2001)NZERO,AZERO
 2001 FORMAT(16X,F10.1,1X,F9.6)
      NOP=NOP+1
      RETFAC = DDNRET/DADN
      IF(DADN .EQ. 0.0) RETFAC = 0.0
      WRITE(6,2100) IFLT,ISEG,J4,CYC,A,DLKEFF,KMXEFF,DDNRET,RETFAC
      WRITE(1,2100) IFLT,ISEG,J4,CYC,A,DLKEFF,KMXEFF,DDNRET,RETFAC
      RETURN
 2000 FORMAT(2X*FLT  MSN  LYR   CYCLES*7X*A*4X
     /*DELTA K    K MAX*5X*DA/DN    RETARD* )
 2100 FORMAT(I5,2X,I3,2X,I3,1X,F10.1,1X,F9.6,1X,
     /F7.2,1X,F7.2,1X,E10.3,1X,F7.3)
      END
      SUBROUTINE INPUT(ICASE,ISPEC,IRSTRT)
C
C     READS LABELED SECTIONS OF DATA DECK IN ANY ORDER.
C     PRINTS OUT PROBLEM AND SOLUTION DESCRIPTION.
C
      COMMON/DATA/ EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AMAX,NZERO
      INTEGER EQN
      REAL NZERO
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /              KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /             NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
```

```
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /               AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /               BTABL2(100),ASTART(10),ASTOP(10)
      DIMENSION CARD(18),TITLE(18),SEGTTL(18,50)
      REAL LODLAB(3)
      REAL LABEL(3)
      REAL SPCTRM(3),ENDLDS(3),IEND(3),LTITLE(3)
      REAL TAG(2)
      REAL EQNS(3),MATL(3),LIMITS(3),ANAL(3),LOADS(3),END(3)
      REAL DADN(3),WALK(3)
      REAL EQUAT(3),DELK(3)
      REAL FORMAN(3),PARIS(3),NASAL(3),BLANK(3),SIGMOID(3)
      REAL SURF(3),RET(3),BETAL(3),BETAE(3),SIGMAS(3),DELTAS(3)
      REAL AMEANS(3),ENDSPC(3),TRSHLD(3),LRSTRT(3),KPRT(3)
C      THE NEXT 3 CARDS ADDED FOR CLOSUR
      COMMON/CLOS/CF,CCOEF,CFEXP,B,BOL,NSAT
      REAL NSAT
      COMMON/CLOSIC/SC,SPEAK,PRVMX,APEAK,SC1,SC2,SC3,PRVMN
      DATA SPCTRM   /4HSPEC,4HTRUM,1H /,ENDLDS   /3HEND,4HLOAD,1HS/
     +    ,IEND    /3HEND,4HDATA,1H /,LTITLE   /4HTITL,1HE,1H /
      DATA EQNS    /4HEQUA,4HTION,1H /,MATL   /4HMATE,4HRIAL,1H /
     +    ,LIMITS   /4HLIMI,2HTS,1H /,ANAL   /4HANAL,4HYSIS,1H /
     +    ,LOADS   /4HLOAD,1HS,1H /,END   /3HEND,2*1H /
      DATA DADN    /4HDA/D,1HN,1H /,WALK   /4HWALK,2HER,1H /
      DATA FORMAN   /4HFORM,2HAN,1H /,PARIS   /4HPARI,1HS,1H /
     +    ,NASAL   /4HNASA,2*1H /,BLANK   /3*1H /,SIGMOID/4HSIGM,3HOID,1
     +H /
      DATA SURF    /4HSURF,3HACE,1H /,RET    /4HRETA,2HRD,1H /
     +    ,BETAL   /4HBETA,2*1H /,BETAE   /3HEND,2*1H /
     +    ,SIGMAS   /4HMAX-,3HMIN,1H /,DELTAS   /4HR-DE,4HLTA ,1H /
      DATA AMEANS   /4HMEAN,4H-ALT,1H /,ENDSPC   /3HEND,4HSPEC,1HR/
     +    ,TRSHLD   /4HTHRE,4HSHOL,1HD /,LRSTRT   /4HREST,3HART,1H /
     +    ,KPRT    /4HPRIN,1HT,1H /
      ISTOP = 0
      IF(ISPEC.GT.1) GO TO 6
C       INITIAL CONDITIONS FOR CLOSUR
          SC = 0.
          SC1 = 0.
          SC2 = 0.
          SC3 = 0.
          SPEAK = 0.
          PRVMX = 0.
          PRVMN = 0.
          APEAK = 0.
C       END OF INITIAL CONDITIONS FOR CLOSUR
      IRSTRT = 0
      NORTRD = 1
      EQN = 1
      NASA=0
      NPTS=0
      NPTS2=0
      ISURF=0
      MODEL=0
      J1PR = 1
      J2PR = 0
      J3PR = 0
      J4PR = 0
      J5PR = 0
      RATIO = 1.0
      RCUT = 1.
      KSUBC = 68000.
      KSUBQ = KSUBC
      C = 5.0E-13
      SMALLN = 3.0
      DELKTH = 0.
      RMULT = 0.
      DO 5 J=1,10
      BETA(J) = 0.0
    5 IBETA(J) = 0.0
    6 READ(5,1000) LABEL
      IF(EOF(5)) 9999,7
    7 WRITE(6,2900) ICASE,ISPEC
      IF(IRSTRT.NE.0) WRITE(6,3400)
      GO TO 10
    1 READ(5,1000)LABEL
      IF(EOF(5)) 9998,10
   10 IF( LABEL(1) .EQ. LTITLE(1) .AND. LABEL(2) .EQ. LTITLE(2)
     +                        .AND. LABEL(3) .EQ. LTITLE(3) ) GO TO 100
      IF( LABEL(1) .EQ. KPRT(1) .AND. LABEL(2) .EQ. KPRT(2)
     +                      .AND. LABEL(3) .EQ. KPRT(3) ) GO TO 150
      IF( LABEL(1) .EQ. EQNS(1) .AND. LABEL(2) .EQ. EQNS(2)
     +                      .AND. LABEL(3) .EQ. EQNS(3) ) GO TO 200
      IF( LABEL(1) .EQ. MATL(1) .AND. LABEL(2) .EQ. MATL(2)
     +                      .AND. LABEL(3) .EQ. MATL(3) ) GO TO 300
      IF( LABEL(1) .EQ. TRSHLD(1) .AND. LABEL(2) .EQ. TRSHLD(2)
     +                      .AND. LABEL(3) .EQ. TRSHLD(3) ) GO TO 350
      IF( LABEL(1) .EQ. LIMITS(1) .AND. LABEL(2) .EQ. LIMITS(2)
     +                      .AND. LABEL(3) .EQ. LIMITS(3) ) GO TO 400
      IF( LABEL(1) .EQ. ANAL(1) .AND. LABEL(2) .EQ. ANAL(2)
     +                      .AND. LABEL(3) .EQ. ANAL(3) ) GO TO 500
      IF( LABEL(1) .EQ. LOADS(1) .AND. LABEL(2) .EQ. LOADS(2)
     +                      .AND. LABEL(3) .EQ. LOADS(3) ) GO TO 600
      IF( LABEL(1) .EQ. SPCTRM(1) .AND. LABEL(2) .EQ. SPCTRM(2)
     +                      .AND. LABEL(3) .EQ. SPCTRM(3) ) GO TO 660
      IF( LABEL(1) .EQ. LRSTRT(1) .AND. LABEL(2) .EQ. LRSTRT(2)
     +                      .AND. LABEL(3) .EQ. LRSTRT(3) ) GO TO 680
      IF( LABEL(1) .EQ. IEND(1) .AND. LABEL(2) .EQ. IEND(2)
     +                      .AND. LABEL(3) .EQ. IEND(3) ) GO TO 700
      WRITE(6,9020) LABEL
      ISTOP = 1
      GO TO 1
  100 READ(5,1010) NTITLE
      DO 110 I=1,NTITLE
      READ(5,1000) CARD
      WRITE(6,2005) CARD
      WRITE(1,2005) CARD
  110 CONTINUE
      GO TO 1
  150 READ(5,1010) J1PR,J2PR,J3PR,J4PR,J5PR
      GO TO 1
  200 READ(5,1002) EQUAT,DELK
      IF( EQUAT(1) .EQ. FORMAN(1) .AND. EQUAT(2) .EQ. FORMAN(2)
     +                      .AND. EQUAT(3) .EQ. FORMAN(3) )EQN = 1
      IF( EQUAT(1) .EQ. PARIS(1) .AND. EQUAT(2) .EQ. PARIS(2)
     +                      .AND. EQUAT(3) .EQ. PARIS(3) )EQN = 2
      IF( EQUAT(1) .EQ. DADN(1) .AND. EQUAT(2) .EQ. DADN(2)
     +                      .AND. EQUAT(3) .EQ. DADN(3) )EQN = 3
      IF( EQUAT(1) .EQ. WALK(1) .AND. EQUAT(2) .EQ. WALK(2)
     +                      .AND. EQUAT(3) .EQ. WALK(3) )EQN = 4
      IF( EQUAT(1) .EQ. SIGMOID(1) .AND. EQUAT(2) .EQ. SIGMOID(2)
     +                      .AND. EQUAT(3) .EQ. SIGMOID(3) )EQN = 5
      IF( DELK(1) .EQ. NASAL(1) .AND. DELK(2) .EQ. NASAL(2)
     +                      .AND. DELK(3) .EQ. NASAL(3) )NASA = EQN
      IF(NASA.NE.0) GO TO 260
      GO TO (210,220,230,240,250),EQN
  210 WRITE(6,2010)
      GO TO 1
  220 WRITE(6,2020)
      GO TO 1
  230 WRITE(6,3030)
      GO TO 1
  240 CONTINUE
      WRITE(6,3050)
      GO TO 1
  250 CONTINUE
      WRITE(6,3070)
      GO TO 1
  260 GO TO (270,280,285,290,295), NASA
  270 WRITE(6,2030)
      GO TO 1
  280 WRITE(6,2040)
      GO TO 1
  285 WRITE(6,3040)
      GO TO 1
  290 CONTINUE
      WRITE(6,3060)
      GO TO 1
  295 CONTINUE
  300 READ(5,1000) MATID
      WRITE(6,2050) MATID
      CALL CNDIN(EQN)
      GO TO 1
  350 READ(5,1020) DELKTH,RMULT
      WRITE(6,2260) DELKTH,RMULT
      GO TO 1
```

```
400 READ(5,1020) AZERO,AMAX,NZERO,RCUT
    IF(RCUT.EQ.0.) RCUT =1.
    IF(AMAX.EQ.0.) GO TO 410
    WRITE(6,2070) AZERO,AMAX,NZERO
    WRITE(6,2075) RCUT
    GO TO 1
410 AMAX=1.0E+50
    WRITE(6,2080) AZERO,NZERO
    WRITE(6,2075) RCUT
    GO TO 1
500 READ(5,1070) LABEL,CON1,CON2,CON3,CON4,CON5,CON6
    IF( LABEL(1) .EQ. SURF(1) .AND. LABEL(2) .EQ. SURF(2)
   +                       .AND. LABEL(3) .EQ. SURF(3) ) GO TO 510
    IF( LABEL(1) .EQ. RET(1) .AND. LABEL(2) .EQ. RET(2)
   +                       .AND. LABEL(3) .EQ. RET(3) ) GO TO 520
    IF( LABEL(1) .EQ. BETAL(1) .AND. LABEL(2) .EQ. BETAL(2)
   +                       .AND. LABEL(3) .EQ. BETAL(3) ) GO TO 530
    IF( LABEL(1) .EQ. BETAE(1) .AND. LABEL(2) .EQ. BETAE(2)
   +                       .AND. LABEL(3) .EQ. BETAE(3) ) GO TO 1
    WRITE(6,9020) LABEL
    ISTOP = 1
    GO TO 1
510 ISURF=1
    CZERO=CON1
    THICK=CON2
    RATIO=AZERO/(2.*CZERO)
    SMALLK = SQRT((CZERO**2 - AZERO**2) / CZERO**2)
    CKSQD = 1.0 - SMALLK**2
    CALL CELI2(PHI,SMALLK,1.0,CKSQD,IER)
    WRITE(6,2090) RATIO,THICK,PHI
    IF(IER.EQ.0) GO TO 500
    WRITE(6,9030)
    ISTOP = 1
    GO TO 500
520 MODEL=CON1+0.5
    PLSTRN=CON2+1.5
    NORTRD = CON4 + 0.5
    OVLD = CON5
    ASUBP = CON6
    IF(NORTRD.NE.0) WRITE(6,2170)
    GO TO (521,522,523,524,525),MODEL
521 SMALLM=CON3
    WRITE(6,2100) SMALLM
    GO TO 526
522 WRITE(6,2105)
    OLMAX = CON3
    IF(OLMAX.NE.0.) WRITE(6,2106) OLMAX
    GO TO 526
523 CONTINUE
    WRITE(6,2107)
    READ(5,1020) CF,CCOEF,CFEXP,B,BOL,NSAT
    WRITE(6,2108) CCOEF,CF,CCOEF,CFEXP,B,BOL,NSAT
```

```
      RETARD = 1
        GO TO 526
  524 CONTINUE
  525 WRITE(6,3000)
      MODEL=2
  526 GO TO (527,528),PLSTRN
  527 WRITE(6,2180)
      PLSTRN = PLSTRN - 1
      GO TO 500
  528 WRITE(6,2190)
      PLSTRN = PLSTRN - 1
      GO TO 500
  530 I = CON1 + 0.5
      IBETA(I)=I
      GO TO(531,532,533,534,535,536,537,538,539),I
  531 BETA(I)=CON2
      ASTART(I)=CON3
      ASTOP(I)=CON4
      IF(ASTART(I).EQ.0.) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.) ASTOP(I) = 1.E50
      WRITE(6,2110) BETA(1),ASTART(I),ASTOP(I)
      GO TO 500
  532 BETA(I)=CON2
      ASTART(I)=CON3
      ASTOP(I)=CON4
      IF(ASTART(I).EQ.0.) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.) ASTOP(I) = 1.E50
      WRITE(6,2120) BETA(2),ASTART(I),ASTOP(I)
      GO TO 500
  533 BETA(I)=CON3
      NPTS=CON2+0.5
      ASTART(I)=CON4
      ASTOP(I)=CON5
      IF(ASTART(I).EQ.0.) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.) ASTOP(I) = 1.E50
      IF(NPTS.LE.0.OR.NPTS.GT.100) GO TO 540
      READ(5,1000) CARD
      READ(5,1040) (AOVERB(J),BTABLE(J),J=1,NPTS)
      WRITE(6,2130) BETA(I),ASTART(I),ASTOP(I),CARD,
     1             (AOVERB(J),BTABLE(J),J=1,NPTS)
      GO TO 500
  534 BETA(I)=CON3
      NPTS2=CON2+0.5
      ASTART(I)=CON4
      ASTOP(I)=CON5
      IF(ASTART(I).EQ.0.) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.) ASTOP(I) = 1.E50
      IF(NPTS2.LE.0.OR.NPTS2.GT.100) GO TO 540
      READ(5,1000) CARD
      READ(5,1040) (AOVRB2(J),BTABL2(J),J=1,NPTS2)
      WRITE(6,2140) BETA(I),ASTART(I),ASTOP(I),CARD,
     1             (AOVRB2(J),BTABL2(J),J=1,NPTS2)
```

```
      GO TO 500
  535 CONTINUE
      BETA(I) = CON2
      ASTART(I) = CON3
      ASTOP(I) = CON4
      IF(ASTART(I).EQ.0.0) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.0) ASTOP(I) = 1.0E50
      WRITE(6,2200) BETA(I),ASTART(I),ASTOP(I)
      GO TO 500
  536 CONTINUE
      BETA(I) = CON2
      ASTART(I) = CON3
      ASTOP(I) = CON4
      IF(ASTART(I).EQ.0.0) ASTART(I) = AZERO
      IF(ASTOP(I).EQ.0.0) ASTOP(I) = 1.0E50
      WRITE(6,2205) BETA(I),ASTART(I),ASTOP(I)
      GO TO 500
  537 CONTINUE
          BETA(I) = CON2
          THICK = CON3
          ASTART(I) = CON4
          ASTOP(I) = CON5
        IF ( ASTART(I) .EQ. 0.0 ) ASTART(I) = AZERO
        IF ( ASTOP(I) .EQ. 0.0 ) ASTOP(I) = 1.E50
      WRITE(6,2206) BETA(I),THICK,ASTART(I),ASTOP(I)
      GO TO 500
  538 CONTINUE
          BETA(I) = CON2
          THICK = CON3
          ASTART(I) = CON4
          ASTOP(I) = CON5
        IF ( ASTART(I) .EQ. 0.0 ) ASTART(I) = AZERO
        IF ( ASTOP(I) .EQ. 0.0 ) ASTOP(I) = 1.E50
      WRITE(6,2207) BETA(I),THICK,ASTART(I),ASTOP(I)
      GO TO 500
  539 CONTINUE
          BETA(I) = CON2
          THICK = CON3

          ASTART(I) = CON4
          ASTOP(I) = CON5
        IF ( ASTART(I) .EQ. 0.0 ) ASTART(I) = AZERO
        IF (ASTOP(I) .EQ. 0.0 ) ASTOP(I) = 1.E50
      WRITE(6,2208) BETA(I),THICK,ASTART(I),ASTOP(I)
      GO TO 500
  540 WRITE(6,9000) IBETA(I)
      IBETA(I)=0
      GO TO 500
  600 READ(5,1050) NBLKS,LPRT,(TITLE(I),I=1,18)
      WRITE(6,2000)(TITLE(I),I=1,18),NBLKS
      IF(LPRT.LT.0) GO TO 1
      ISEG = 0
  605 ISEG = ISEG + 1
      READ(5,1002) LODLAB,(SEGTTL(I,ISEG),I=1,18)
      LYR=1
      IF( LODLAB(1) .EQ. SIGMAS(1) .AND. LODLAB(2) .EQ. SIGMAS(2)
     +                        .AND. LODLAB(3) .EQ. SIGMAS(3) ) GO TO 610
      IF( LODLAB(1) .EQ. DELTAS(1) .AND. LODLAB(2) .EQ. DELTAS(2)
     +                       .AND. LODLAB(3) .EQ. DELTAS(3) ) GO TO 620
      IF( LODLAB(1) .EQ. AMEANS(1) .AND. LODLAB(2) .EQ. AMEANS(2)
     +                       .AND. LODLAB(3) .EQ. AMEANS(3) ) GO TO 630
      IF( LODLAB(1) .EQ. ENDLDS(1) .AND. LODLAB(2) .EQ. ENDLDS(2)
     +                       .AND. LODLAB(3) .EQ. ENDLDS(3) ) GO TO 662
      WRITE(6,3020) LODLAB
      ISTOP = 1
      GO TO 1
C
C     LOAD SPECTRUM INPUT AS MAX AND MIN STRESSES
C
  610 READ(5,1030) TAG,SMAX(LYR,ISEG),SMIN(LYR,ISEG),CYCLES(LYR,ISEG)
      IF( TAG(1) .EQ. END(1) .AND. TAG(2) .EQ. END(2) ) GO TO 650
C
C     PRESENT PROGRAM DOES NOT CONSIDER COMPRESSIVE LOADS
C
C          EXCEPT FOR CLOSURE MODEL WHICH DOES
C
        IF ( MODEL .EQ. 3 ) GO TO 618
      IF(SMAX(LYR,ISEG).LT.0.) SMAX(LYR,ISEG)=0.
      IF(SMIN(LYR,ISEG).LT.0.) SMIN(LYR,ISEG) = 0.
  618 LYR=LYR+1
      GO TO 610
  620 IF(LPRT.EQ.0)
     1WRITE(6,3010) ISEG,(SEGTTL(I,ISEG),I=1,18)
      IF(LPRT.EQ.0) WRITE(6,2210)
  625 READ(5,1030) TAG,DELSIG,R,CYCLES(LYR,ISEG)
      IF( TAG(1) .EQ. END(1) .AND. TAG(2) .EQ. END(2) ) GO TO 650
      IF(LPRT.EQ.0)
     1WRITE(6,2220) LYR,TAG,DELSIG,R,CYCLES(LYR,ISEG)
C
C     PRESENT PROGRAM DOES NOT CONSIDER COMPRESSIVE LOADS
C
      IF(DELSIG.LT.0.) DELSIG=0.
      IF(R.LT.0.)R=0.
      SMAX(LYR,ISEG)=DELSIG/(1.-R)
      SMIN(LYR,ISEG)=SMAX(LYR,ISEG)-DELSIG
      LYR=LYR+1
      GO TO 625
  630 IF(LPRT.EQ.0)
     1WRITE(6,3010) ISEG,(SEGTTL(I,ISEG),I=1,18)
      IF(LPRT.EQ.0) WRITE(6,2230)
  635 READ(5,1030) TAG,SMEAN,SALT,CYCLES(LYR,ISEG)
      IF( TAG(1) .EQ. END(1) .AND. END(2) .EQ. TAG(2)) GO TO 650
      IF(LPRT.EQ.0)
     1WRITE(6,2240) LYR,TAG,SMEAN,SALT,CYCLES(LYR,ISEG)
```

```
C
C     PRESENT PROGRAM DOES NOT CONSIDER COMPRESSIVE LOADS
C
      SMAX(LYR,ISEG)=SMEAN+SALT
      IF(SMAX(LYR,ISEG).LT.0.) SMAX(LYR,ISEG)=0.
      SMIN(LYR,ISEG)=SMEAN-SALT
      IF(SMIN(LYR,ISEG).LT.0.) SMIN(LYR,ISEG)=0.
      LYR=LYR+1
      GO TO 635
  650 NLYRS(ISEG) = LYR-1
      GO TO 605
C
C     READ IN COMPOSITION OF SPECTRUM
C
  660 READ(5,1010)NSEGS,IPRT
      READ(5,1060) (IBLKS(I),ISEGS(I),I=1,NSEGS)
      IF(IPRT.EQ.0) WRITE(6,2250) NSEGS
      IFLTS = 0
      DO 661 I=1,NSEGS
      IFLTS = IFLTS + IBLKS(I)
      IF(IPRT.EQ.0) WRITE(6,2255) I,IBLKS(I),ISEGS(I),IFLTS
  661 CONTINUE
      GO TO 1
  662 NNSEG = ISEG - 1
      IF(LPRT.NE.0) GO TO 1
      DO 670 ISEG =1,NNSEG
      WRITE(6,2270) ISEG,(SEGTTL(I,ISEG),I=1,18)
      LYRS = NLYRS(ISEG)
      TOTCYC = 0.0
      DO 665 J2=1,LYRS
      TOTCYC = TOTCYC + CYCLES(J2,ISEG)
      WRITE(6,2280) J2,SMAX(J2,ISEG),SMIN(J2,ISEG),CYCLES(J2,ISEG),
     1              TOTCYC
  665 CONTINUE
  670 CONTINUE
      GO TO 1
  680 READ(5,1060) IRSTRT
      ICHECK = IRSTRT + 1
      GO TO (1,685,685,690,695,695),ICHECK
  685 WRITE(6,2150)
      GO TO 1
  690 WRITE(6,2160)
      GO TO 1
  695 WRITE(6,2165)
      GO TO 1
  700 WRITE(6,2015) (TITLE(I),I=1,18)
      RETURN
 9998 IF(IRSTRT .NE. 0) WRITE(6,9010)
 9999 STOP
C
 1000 FORMAT(20A4)
 1002 FORMAT(2A4,A2,17A4,A2)
```

```
1010 FORMAT(16I5)
1020 FORMAT(8E10.0)
1030 FORMAT(A4,A1,7E10.0)
1040 FORMAT(2E10.0)
1050 FORMAT(2I5,17A4,A2)
1060 FORMAT(2I5)
1070 FORMAT(2A4,A2,7E10.0)
2000 FORMAT(1H0, 1X,17A4,A2// 5X,I6,19H BLOCKS IN SPECTRUM)
2005 FORMAT( 2X,17A4,A2)
2010 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING FORMAN'S EQUATI
    1ON    / 4X,44HDA/DN=C*(DELTA K)**N/((1-R)*KSUBC-DELTA K)
    2     / 5X,51HWHERE K IS OF THE FORM .... K=SIGMA*SQRT(PI*A)*BETA)
2015 FORMAT(//1X,29(1H*),12HEND OF INPUT,29(1H*)/////1H1,1X,70(1H$)/
    /1X,26(1H*),18HCRACKS IV ANALYSIS,26(1H*)/1X,17A4,A2/
    /1X,70(1H$)//)
2020 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING PARIS' EQUATION
    1    /16X,21HDA/DN=C*(DELTA K)**N / 5X,51HWHERE K IS OF THE FORM .
    2.. K=SIGMA*SQRT(PI*A)*BETA )
2030 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING FORMAN'S EQUATI
    1ON    / 4X,44HDA/DN=C*(DELTA K)**N/((1-R)*KSUBC-DELTA K)
    2    / 5X,51HWHERE K IS OF THE FORM .... K=SIGMA*SQRT(A)*BETA    )
2040 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING PARIS' EQUATION
    1    /16X,21HDA/DN=C*(DELTA K)**N / 5X,51HWHERE K IS OF THE FORM .
    2.. K=SIGMA*SQRT(A)*BETA     )
2050 FORMAT(1H0, 1X,20A4)
2070 FORMAT(1H0, 1X,27HINITIAL HALF CRACK LENGTH = E16.8/2X,35HMAXIMUM
    1HALF CRACK LENGTH ALLOWED = E16.8/1H0, 1X,22HINITIAL CYCLE NUMBER
    2   ,F11.2 )
2075 FORMAT(1H0, 1X,11HR CUTOFF = ,F6.3)
2080 FORMAT(1H0, 1X,27HINITIAL HALF CRACK LENGTH = E16.8 /1H0, 1X,22HIN
    1ITIAL CYCLE NUMBER = ,F11.2 )
2090 FORMAT(1H0, 1X,34HSURFACE FLAW ANALYSIS WITH A/2C OF F5.2 / 5X,21H
    1MATERIAL THICKNESS IS ,F8.5/ 5X,19HSHAPE FACTOR PHI = ,F8.5)
2100 FORMAT(1H0, 1X,41HWHEELER'S RETARDATION MODEL WITH SMALLM = ,F6.3)
2105 FORMAT(1H0,1X,*WILLENBORG RETARDATION MODEL*)
2106 FORMAT(1H0,1X,*GALLAGHER-MODIFIED WILLENBORG RETARDATION MODEL */
    2X*WHERE...*// 5X,27HPHI = (1-THRESHOLD/K MAX)/(,F5.2,3H-1))
C       FORMATS FOR CLOSUR INPUT/OUTPUT DATA
2107 FORMAT(1H0, 1X,18HCLOSURE MODEL WITH)
2108 FORMAT(1H0, 1X,16HCLOSURE FACTOR =,F6.4,3H+ (,F6.4,1H-,F6.4,
    1      10H)*(1.-R)**,F6.4,/ 5X,33HEXPONENT FOR DECREASING CLOSURE =,
    /      F6.3,/
    2       5X,34HEFFECTIVENESS AFTER ONE OVERLOAD =,F6.4/
    3       5X,36HNUMBER OF OVERLOADS FOR SATURATION =,F6.0)
2110 FORMAT(1H0, 1X,40HCORRECTION FACTOR BETA(1) IS A CONSTANT  / 5X,9H
    1BETA(1) = ,E16.8/ 5X,16HAPPLIED FROM A= ,E16.8,7H TO A = ,E16.8)
2120 FORMAT(1H0, 1X,54HCORRECTION FACTOR BETA(2) IS FINITE WIDTH CORREC
    1TION    /5X28HBETA(2) = SQRT(SEC(PI*A/B))  /
    25X*WHERE THE EFFECTIVE PLATE WIDTH W = *E16.8/
    35X*APPLIED FROM A = *E16.8,* TO A = *E16.8)
2130 FORMAT(1H0, 1X,54HCORRECTION FACTOR BETA(3) IS A TABULAR FUNCTION
    1OF A/L  / 5X,10HWHERE L = ,E16.8 / 5X,16HAPPLIED FROM A = ,E16.8,7
```

```
     2H TO A = ,E16.8// 1X,18A4// 8X,3HA/L,16X,7HBETA(3)/( 5X,E15.8,5X, 006600
     3E15.8))                                                           006610
2140 FORMAT(1HO, 5X,55HCORRECTION FACTOR BETA(4) IS A TABULAR FUNCTION 006620
     1OF A/L1  / 5X,10HWHERE L = E16.8/ 5X,16HAPPLIED FROM A = ,E16.8,7H006630
     2 TO A = ,E16.8// 1X,18A4// 8X,3HA/L,16X,7HBETA(4)/( 5X,E15.8,5X,E1006640
     35.8))                                                             006650
2150 FORMAT(1HO, 1X,35HTHIS CASE WILL BE RESTARTED ON-LINE)             006660
2160 FORMAT(1HO, 1X,36HTHIS CASE WILL BE RESTARTED OFF-LINE      /      006670
     1       48H  RESTART DATA WILL BE WRITTEN ON LOGICAL UNIT 7)       006680
2165 FORMAT(1HO,1X,36HTHIS CASE WILL BE RESTARTED OFF-LINE     /        006690
     1          48H  RESTART DATA WILL BE READ FROM LOGICAL UNIT 7   )  006700
2170 FORMAT(1HO, 1X,40HAUTOMATIC UNRETARDED SOLUTION SUPPRESSED)        006710
2180 FORMAT(1HO, 1X,41HPLANE STRESS YIELD ZONE CONDITION ASSUMED )      006720
2190 FORMAT(1HO, 1X,41HPLANE STRAIN YIELD ZONE CONDITION ASSUMED )      006730
2200 FORMAT(1HO, 1X,71HCORRECTION FACTOR BETA(5) IS UNIAXIAL BOWIE SOLU006740
     1TION FOR A SINGLE CRACK/ 5X,35HFROM A CIRCULAR HOLE OF RADIUS R = 006750
     2E16.8 / 5X,16HAPPLIED FROM A = ,E16.8,8H TO A = ,E16.8)           006760
2205 FORMAT(1HO, 1X,68HCORRECTION FACTOR BETA(6) IS UNIAXIAL BOWIE SOLU006770
     1TION FOR TWO CRACKS /5X,35HFROM A CIRCULAR HOLE OF RADIUS R = ,   006780
     2E16.8 / 5X,17HAPPLIED FROM A = ,E16.8,8H TO A = ,E16.8)           006790
2206 FORMAT(1HO, 1X,58HCORRECTION FACTOR BETA(7) IS ASTM COMPACT TENSIO006800
     1N SPECIMEN/ 5X,15HWITH A WIDTH OF,E16.8/                          006810
     /       5X,16HAND THICKNESS OF,E16.5/                              006820
     2        5X,17HAPPLIED FROM A = ,E16.8,8H TO A = ,E16.8)           006830
2207 FORMAT(1HO, 1X,61HCORRECTION FACTOR BETA(8) IS GRUMMAN COMPACT TEN006840
     1SION SPECIMEN/ 5X,15HWITH A WIDTH OF,E16.8/                       006850
     /       5X,16HAND THICKNESS OF,E16.5/                              006860
     2        5X,17HAPPLIED FROM A = ,E16.8,8H TO A = ,E16.8)           006870
2208 FORMAT(1HO, 1X,71HCORRECTION FACTOR BETA(9) IS ASTM  E647-83 FOR A
     1COMPACT TENSION SPEIMEN/ 5X,15HWITH A WIDTH OF,E16.8/
     /       5X,16HAND THICKNESS OF,E16.5/
     2        5X,17HAPPLIED FROM A = ,E16.8,8H TO A = ,E16.8)
2210 FORMAT(2X*LAYER*4X*LABEL*7X*DELTA*9X*R*4X*CYCLES PER*/             006880
     /23X*SIGMA*16X*LAYER*//)                                           006890
2220 FORMAT(3X,I2,5X,2A4,1X,E13.5,1X,F7.4,2X,F10.2)                     006900
2230 FORMAT(2X*LAYER*4X*LABEL*8X*MEAN*6X*ALTERNATING*3X                 006910
     /*CYCLES PER*/23X*STRESS*8X*STRESS*7X*LAYER*//)                    006920
2240 FORMAT(3X,I2,5X,2A4,1X,E13.5,1X,E13.5,2X,F10.2)                    006930
2250 FORMAT(1H1,1X,I6,52H SEGMENT SPECTRUM APPLIED IN THE FOLLOWING SEQ006940
     /UENCE //1X*SEGMENT*11X*FLIGHTS PER*6X*MISSION*11X*CUMULATIVE*/    006950
     /21X*MISSION*28X*FLIGHTS*//)                                       006960
2255 FORMAT(3X,I2,17X,I4,12X,I2,17X,I5)                                 006970
2260 FORMAT(1HO, 1X,19HTHRESHOLD DELTA K =,E17.8,7H (1.0-(,F6.3,4H)*R))006980
2270 FORMAT(1H1, 70(1H*)/ 5X,34HSTRESS SPECTRUM FOR MISSION NUMBER,I3/  006990
     /2X,17A4,A2/1X,70(1H*)//2X*LAYER*5X*MAXIMUM*7X*MINIMUM*7X          007000
     /*CYCLES*6X*CUMULATIVE*/13X*STRESS*8X*STRESS*8X*PER*8X             007010
     /*CYCLES PER*/40X*LAYER*9X*MISSION*//)                             007020
2280 FORMAT(3X,I2,4X,4(F12.3,2X))                                       007030
2900 FORMAT(1H1,70(1H*)/26X,5HCASE ,I2,5X,4HRUN ,I2/1X,70(1H*))         007040
3000 FORMAT(1HO, 70(1H*)/ 5X,66HINACTIVE RETARDATION MODEL CHOSEN.EFFEC007050
     1TIVE STRESS MODEL ASSUMED.  /1X, 70(1H*))                         007060
3010 FORMAT(1H1, 5X,33HINPUT SPECTRUM FOR MISSION NUMBER,I4/ 5X,18A4)   007070
```

```
3030 FORMAT(1H0, 1X,65HCRACK PROPAGATION ANALYSIS USING DIRECT INPUT OF
    1 DA/DN VS.DELTA K / 5X,51H WHERE K IS OF THE FORM ....SIGMA*SQRT(P
    2I*A)*BETA  )
3040 FORMAT(1H0, 1X,65HCRACK PROPAGATION ANALYSIS USING DIRECT INPUT OF
    1 DA/DN VS.DELTA K  / 5X,51HWHERE K IS OF THE FORM .... K=SIGMA*SQR
    2T(A)*BETA     )
3050 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING WALKER'S EQUATI
    1ON/ 7X,35HDA/DN=C*(DELTA K/((1-R)**(1-M)))**N / 5X,52HWHERE K IS
    2OF THE FORM .... K=SIGMA*SQRT(PI*A)*BETA          )
3060 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING WALKER'S EQUATI
    1ON/ 7X,35HDA/DN=C*(DELTA K/((1-R)**(1-M)))**N / 5X,51HWHERE K IS O
    2F THE FORM .... K=SIGMA*SQRT(A)*BETA             )
3070 FORMAT(1H0, 1X,50HCRACK PROPAGATION ANALYSIS USING SIGMOIDAL EQN.
    1  / 7X,60HDA/DN=EXP(B) (DELTA K/DELTA K*)**P (LN(DELTA K/DELTA K*)
    2)**Q/12X,26H (LN(DELTA KC/DELTA K))**D / 5X,51HWHERE K IS OF THE F...
    3ORM .... K=SIGMA*SQRT(PI*A)*BETA            )
3400 FORMAT(1H0,70(1H*)/21X,40HRERUN OF CASE WITH THE FOLLOWING CHANGES
    /  / 1X,70(1H*))
9000 FORMAT(1H0, 70(1H*)/ 5X,51HNUMBER OF POINTS IN TABULAR CORRECTION
    1FACTOR BETA(I1,*) EXCEEDS 100.*/* CORRECTION FACTOR WILL BE IGNORE
    2D.*/1X, 70(1H*) )
9010 FORMAT(1H0, 70(1H*)/ 5X,38HERROR IN DECK SETUP.E-O-F ENCOUNTERED.
    1       /1X, 70(1H*))
9020 FORMAT(1H0, 70(1H*)/ 5X,48HINCORRECT LABEL CARD ENCOUNTERED.LABEL
    +READ WAS ,2A4,A2,1H*/ 1X,64HEXECUTION SUPPRESSED.PROGRAM WILL COMP
    2LETE INPUT DATA PROCESSING/1X, 70(1H*))
9030 FORMAT(1H0, 70(1H*)/ 1X,71HERROR IN CALCULATING PHI.PROGRAM REQUIR
    +ES (AZERO/2(CZERO))SQD .LE.  0.5/ 1X,65HEXECUTION SUPPRESSED. PROG
    +RAM WILL COMPLETE INPUT DATA PROCESSING/1X, 70(1H*))
     END
     SUBROUTINE DELTA(A,DELTAK,KMAX,R)
     COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
     INTEGER RETARD,PLSTRN
     COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
    /              KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
     REAL KSUBC,KSUBQ
     COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
    /               AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
    /               BTABL2(100),ASTART(10),ASTOP(10)
     REAL KMAX,KMIN
     CALL K(SIGMIN,A,KMIN)
     CALL K(SIGMAX,A,KMAX)
     R  =  0.0
     IF( KMAX .NE. 0.0 ) R = KMIN/KMAX
     DELTAK=KMAX-KMIN
     RETURN
     END
     REAL FUNCTION  TRP2(T,X,Y,M)
C
     DIMENSION T(100),Z(4),D(6)
C
     L1=0
```

```
      X1=X
      Y1=Y
      I = T(1) / 1000. + 1.
      J = AMOD(T(1),1000.) + 1.
      L=J*M
      I1=J*3+1
      I2=I*J
      M1=M
      DO 10 K=I1,I2,L
      IF(X1-T(K)) 20,20,10
   10 CONTINUE
      K=I2+1-J
   20 DO 30 L=4,J,M1
      IF (Y1-T(L)) 40,40,30
   30 CONTINUE
      L=J
   40 L1=L1+1
      DO 50 MN=1,3
      N=L+MN-3
      N1=K+(J*(L1-3))+N-1
      D(MN)=T(N)
   50 D(MN+3)=T(N1)
   60 Z(L1)=D(4)+(Y1-D(1))*((D(5)-D(4))/(D(2)-D(1))+(
     1Y1-D(2))/(D(3)-D(1))*((D(6)-D(5))/(D(3)-D(2))
     2-(D(5)-D(4))/(D(2)-D(1))))
      IF (L1-3)40,70,90
   70 DO 80 MN=1,3
      D(MN+3)=Z(MN)
      N1=K+(J*(MN-3))
   80 D(MN)=T(N1)
      L1=4
      Y1=X
      GO TO 60
   90  TRP2=Z(4)
      RETURN
      END
      SUBROUTINE RESTRT(CYC,A,IRSTRT)
      COMMON/DATA/I1(7),R1(3)
      COMMON/RDATA  /I2(3),R2(5)
      COMMON/MDATA   /I3(18),R3(209)
      COMMON/LDATA/R4( 600 ),I4( 112)
      COMMON/STEPS  /I5(8)
      COMMON/CORFAC /I6,R5(3),I7(10),R6(10),I8,R7(200),I9,R8(220)
      COMMON/MKCRVE/R9(100)
      COMMON/OUTPOT/R10(3),I10(2)
      REWIND 7
      INOUT = IRSTRT - 2
      GO TO (100,200,200),INOUT
C
C     WRITE RESTART TAPE
C
  100 WRITE(7) I1,R1,I2,R2,I3,R3
```

```
      WRITE(7) R4,I4,I10
      WRITE(7) I5,I6,R5,I7,R6,I8,R7,I9,R8,R9,R10,CYC,A
      RETURN
C
C     READ RESTART TAPE
C
  200 READ (7) I1,R1,I2,R2,I3,R3
      READ (7) R4,I4,I10
      READ (7) I5,I6,R5,I7,R6,I8,R7,I9,R8,R9,R10,CYC,A
      WRITE(6,2000) CYC,A
      RETURN
 2000 FORMAT(1H0, 70(1H*)/ 5X,43HRESTART TAPE READ. THIS RUN BEGINS AT C
     1YCLE , E16.8/5X,24H WITH A CRACK LENGTH OF ,F9.5/1X, 70(1H*))
      END
      SUBROUTINE CNDIN(IEQN)
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /                 KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,QLMAX
      COMMON/PARIS/C1,SN1,DKCOM,C2,SN2
      COMMON /DIRECT/ NDADN
      COMMON /WALKER/ CWALK,EXPM,EXPN
      COMMON /SIGMOID/DKSTAR,TOUGH,BEE,PEA,QUE,DEE
      REAL KSUBC,KSUBQ
      GO TO (100,300,200,400,500),IEQN
  100 READ(5,1200) C,SMALLN,KSUBC
      WRITE(6,2700) C,SMALLN,KSUBC
      GO TO 600
  200 READ(5,1100) PTS
      NDADN = PTS + 0.5
      READ(5,1100) (CARRAY(I),SNARAY(I),I= 1,NDADN)
      WRITE(6,2000)
      WRITE(6,2100)(CARRAY(I),SNARAY(I),I = 1,NDADN)
      DO 250 J=1,NDADN
  250 SNARAY(J) = ALOG10(SNARAY(J))
      GO TO 600
  300 READ(5,1200) C1,SN1,DKCOM,C2,SN2
      IF(DKCOM.GT.0.) GO TO 350
      C2 = C1
      SN2 = SN1
      WRITE(6,2550) C1,SN1
      GO TO 600
  350 WRITE(6,2600) C1,SN1,DKCOM,C2,SN2,DKCOM
      GO TO 600
  400 READ(5,1200) CWALK,EXPM,EXPN
      WRITE(6,2500) CWALK,EXPM,EXPN
      GO TO 600
  500 READ(5,1200)DKSTAR,TOUGH,BEE,PEA,QUE,DEE
      WRITE(6,2520)BEE,DKSTAR,PEA,DKSTAR,QUE,TOUGH,DEE
 2520 FORMAT(1X,10HDA/DN=EXP(,F12.7,12H)*((DELTA-K/,F12.7,3H)**,F6.2,1H)
     1,1H* / 1X,13H((LN(DELTA-K/,F12.7,4H))**,F6.2,2H)*
     2/,1X,5H((LN(,F12.7,12H/DELTA-K))**,F6.2,1H))
  600 READ(5,1100) KSUBQ,SIGMAY
      WRITE(6,2400) KSUBQ,SIGMAY
```

```
1100 FORMAT(2E10.0)
1200 FORMAT(8E10.0)
2000 FORMAT(1H0, 1X,32HDIRECT INPUT OF DA/DN VS DELTA K//34X,7HDELTA K
    1  ,14X,5HDA/DN )
2100 FORMAT( 5X,2E20.8)
2400 FORMAT(1H0, 1X,7HKSUBQ = ,E15.8,5X,15HYIELD STRESS =  ,E15.8 )
2500 FORMAT(1H0, 1X,3HC =,E16.8,5X,3HM =,F6.4,5X,3HN =,F6.3)
2550 FORMAT(1H0,1X,30HLINEAR PARIS FIT AS FOLLOWS        /
    /       5X,E12.4,11H(DELTA K)**  ,F5.3)
2600 FORMAT(1H0, 1X,29HBILINEAR PARIS FIT AS FOLLOWS/ 5X,E12.4,11H(DELT
    1A K)**,F5.3,22H FOR DELTA K LESS THAN,E16.8/5X,E12.4,11H(DELTA K)*
    2*   ,F5.3,22H FOR DELTA K MORE THAN ,E16.8  )
2700 FORMAT(1H0, 1X,3HC =,E16.8,5X,8HSMALLN =,F6.3,5X,7HKSUBC =,E16.8)
      RETURN
      END
      BLOCK DATA
      COMMON/MKCRVE/ MK(100)
      REAL MK
C               INITIALIZATION VALUES FOR FIRST 84 ELEMENTS OF MK
      DATA    MK/
     1    11006.,0.05,0.10,0.20,0.30,0.40,0.50,
     2         0.0,1.00,1.00,1.00,1.00,1.00,1.00,
     3         0.1,1.01,1.01,1.01,1.01,1.01,1.00,
     4         0.2,1.03,1.03,1.02,1.02,1.01,1.00,
     5         0.3,1.06,1.06,1.04,1.03,1.02,1.00,
     6         0.4,1.12,1.12,1.08,1.05,1.02,1.00,
     7         0.5,1.22,1.18,1.14,1.08,1.03,1.00,
     8         0.6,1.34,1.30,1.22,1.13,1.06,1.01,
     9         0.7,1.48,1.42,1.31,1.20,1.08,1.02,
     A         0.8,1.64,1.57,1.41,1.26,1.13,1.04,
     B         0.9,1.77,1.68,1.50,1.32,1.18,1.08,
     C         1.0,1.84,1.75,1.59,1.38,1.22,1.10/
      END
      SUBROUTINE CELI2(RES,AK,A,B,IER)
C
C     ..................................................................
C
C        SUBROUTINE CELI2
C
C        PURPOSE
C           COMPUTES THE GENERALIZED COMPLETE ELLIPTIC INTEGRAL OF
C           SECOND KIND.
C
C        USAGE
C           CALL CELI2(RES,AK,A,B,IER)
C
C        DESCRIPTION OF PARAMETERS
C           RES   - RESULT VALUE
C           AK    - MODULUS (INPUT)
C           A     - CONSTANT TERM IN NUMERATOR
C           B     - FACTOR OF QUADRATIC TERM IN NUMERATOR
C           IER   - RESULTANT ERROR CODE WHERE
```

```
C                   IER=0  NO ERROR
C                   IER=1  AK NOT IN RANGE -1 TO +1
C
C        REMARKS
C           FOR AK = +1,-1 THE RESULT VALUE IS SET TO 1.E75 IF B IS
C           POSITIVE, TO -1.E75 IF B IS NEGATIVE.
C           SPECIAL CASES ARE
C           K(K) OBTAINED WITH A = 1, B = 1
C           E(K) OBTAINED WITH A = 1, B = CK*CK WHERE CK IS
C           COMPLEMENTARY MODULUS.
C           B(K) OBTAINED WITH A = 1, B = 0
C           D(K) OBTAINED WITH A = 0, B = 1
C           WHERE K, E, B, D DEFINE SPECIAL CASES OF THE GENERALIZED
C           COMPLETE ELLIPTIC INTEGRAL OF SECOND KIND IN THE USUAL
C           NOTATION, AND THE ARGUMENT K OF THESE FUNCTIONS MEANS
C           THE MODULUS.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           NONE
C
C        METHOD
C           DEFINITION
C           RES=INTEGRAL((A+B*T*T)/(SQRT((1+T*T)*(1+(CK*T)**2))*(1+T*T))
C           SUMMED OVER T FROM 0 TO INFINITY).
C           EVALUATION
C           LANDENS TRANSFORMATION IS USED FOR CALCULATION.
C           REFERENCE
C           R.BULIRSCH,  NUMERICAL CALCULATION OF ELLIPTIC INTEGRALS
C           AND ELLIPTIC FUNCTIONS, HANDBOOK SERIES SPECIAL FUNCTIONS,
C           NUMERISCHE MATHEMATIK VOL. 7, 1965, PP. 78-90.
C
C     ..................................................................
C
      IER=0
C
C     TEST RANGE
C
      CK=AK*AK
      IF(CK-1.) 20,20,10
   10 IER=1
      RETURN
C
C     COMPUTE COMPLEMENTARY MODULUS
C
   20 GEO=SQRT(1.0-CK)
      IF(GEO) 70,30,70
C
C     SET RESULT VALUE = OVERFLOW
C
   30 IF(B) 40,60,50
   40 RES=-1.E38
      RETURN
   50 RES=1.E38
      RETURN
   60 RES=A
      RETURN
C
C     COMPUTE INTEGRAL
C
   70 ARI=1.
      AA=A
      AN=A+B
      W=B
   80 W=W+AA*GEO
      W=W+W
      AA=AN
      AARI=ARI
      ARI=GEO+ARI
      AN=W/ARI+AN
C
C     TEST OF ACCURACY

C
      IF(AARI-GEO-1.E-4*AARI) 100,100,90
   90 GEO=SQRT(GEO*AARI)
      GEO=GEO+GEO
      GO TO 80
  100 RES=.78539816*AN/ARI
      RETURN
      END
      SUBROUTINE GRWCRK(CYC,A,DN)
      COMMON/PDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /              NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      EXTERNAL RATE,WHELER,WLNBRG
      SIGMAX = SMAX(J4,ISEG)
      SIGMIN = SMIN(J4,ISEG)
      IF(MODEL.GT.0) GO TO 100
      CALL RK1DES(CYC,A,DN,RATE)
      RETURN
  100 GO TO (110,120,130,140,150),MODEL
  110 CALL YLDZNE(CYC,A,DN,WHELER)
      GO TO 200
  120 CALL YLDZNE(CYC,A,DN,WLNBRG)
      GO TO 200
  130 CONTINUE
      CALL CLOSUR(CYC,A,DN)
        GO TO 200
  140 CONTINUE
  150 CONTINUE
  200 RETURN
      END
      SUBROUTINE TRANS(A,ATRANS,CYCTR)
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /                NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /                 AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /                 BTABL2(100),ASTART(10),ASTOP(10)
      ATRANS = THICK
      IF(A.LT.ATRANS)RETURN
  100 AEFF = ATRANS/(2.*RATIO)
      A = AEFF
      ISTOP = 2
      WRITE(6,1000) A,CYCTR
 1000 FORMAT(1H0, 70(1H*)/ 5X,55HTRANSITION TO A THRU CRACK OF EFFECTIVE
     1 LENGTH, AEFF = ,F9.5,4H AT ,F12.2,7H CYCLES/1X, 70(1H*))
      RETURN
      END
      SUBROUTINE RATE (CYCLE,A,DADN)
      COMMON/DATA/ EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AMAX,NZERO
      INTEGER EQN
      REAL NZERO
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /                 KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /                NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /                 AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /                 BTABL2(100),ASTART(10),ASTOP(10)
      COMMON/OUTPOT/ KMAX,KMAXA,DELTAK,IFLT,DADNPR
      COMMON/PARIS/C1,SN1,DKCOM,C2,SN2
      COMMON/DIRECT/ NDADN
      COMMON /WALKER/ CWALK,EXPM,EXPN
      COMMON /SIGMOID/DKSTAR,TOUGH,BEE,PEA,QUE,DEE
      REAL KMAX
      REAL KMAXA
      CALL DELTA(A,DELTAK,KMAX,R)
      IF(ISTOP .NE. 0)GO TO 575
      IF( R. GE. RCUT) R = RCUT
      CALL K(SMAX(J4,ISEG),A,KMAXA)
      IF(DELTAK.LE.0.0) GO TO 300
      IF(ISURF.EQ.0) GO TO 50
      CALL TRANS(A,ATRANS,CYCLE)
      IF(A.LT.ATRANS) GO TO 50
      RETURN
   50 THRSLD = DELKTH*(1.0-RMULT*R)
      IF (DELTAK.LE.THRSLD) GO TO 300
      GO TO (100,200,230,240,250),EQN
  100 DENOM = (1.0 - R) * KSUBC - DELTAK
      IF(KMAXA.GE.KSUBQ) GO TO 400
      IF(DENOM.LE.0.0) GO TO 525
      DADN=(C*(DELTAK)**SMALLN)/DENOM
      RETURN
  200 DELKC = KSUBQ - KMAXA
      IF(DELKC.LE.0.0) GO TO 400
      C = C1
      SMALLN = SN1
      IF(DELTAK.GE.DKCOM) C = C2
      IF(DELTAK.GE.DKCOM) SMALLN = SN2
      DADN=C*(DELTAK)**SMALLN
      RETURN
  230 DADN = TBLKUP(CARRAY,SNARAY,NDADN,100,DELTAK)
      DADN = 10.**DADN
      IF(KMAXA.GE.KSUBQ) GO TO 400
      RETURN
  240 IF(KMAXA.GE.KSUBQ) GO TO 400
      DADN=CWALK*(DELTAK/((1.-R)**(1.-EXPM)))**EXPN
      RETURN
  250 IF (KMAXA .GE. KSUBQ) GO TO 400
      TOUG=TOUGH*(1.-.4)
      IF (DELTAK .GE. TOUG) GO TO 260
      XKSTAR=DKSTAR*(1.-.4)
      IF (DELTAK .LE. XKSTAR) GO TO 300
      DADN=EXP(BEE)*((DELTAK/XKSTAR)**PEA)*((ALOG(DELTAK/XKSTAR))**QUE)
     1*((ALOG(TOUG/DELTAK))**DEE)
      RETURN
  260 ISTOP=1
      WRITE(6,270)
  270 FORMAT(1H0, 70(1H*)/ 5X,50HDELTA-K EXCEEDS THE TOUGHNESS PROB. IS
     1 TERMINATED/1X,70(1H*)/1H0, 1X,26HLAST CALCULATED VALUES ARE///
     2)
      GO TO 575
  300 DADN=0.0
      RETURN
  400 ISTOP = 1
      WRITE(6,500)
  500 FORMAT(1H0, 70(1H*)/ 5X,46HKMAX APPLIED EXCEEDS KSUBQ. PROBLEM TER
     1MINATED/1X, 70(1H*)/1H0, 1X,26HLAST CALCULATED VALUES ARE///)
      GO TO 575
  525 WRITE(6,550)
  550 FORMAT(1H0, 70(1H*)/ 5X,49HDELTA K EXCEEDS (1-R)KSUBC. PROBLEM IS
     1TERMINATED/1X,70(1H*)/1H0, 1X,26HLAST CALCULATED VALUES ARE///)
       ISTOP = 1
  575 CONTINUE
      WRITE(6,600) J1,J2,ISEG,IFLT,J4,CYCLE,A,KMAXA,KMAX,DELTAK,DADN
  600 FORMAT( 5X,18HBLOCK IN SPECTRUM ,I4/
     /          5X,18HSEGMENT NUMBER     ,I4/
     /          5X,18HMISSION NUMBER     ,I4/
     /          5X,18HFLIGHT NUMBER      ,I6/
     /          5X,18HLAYER IN MISSION   ,I4/
     5          5X,18HACCUMULATED CYCLES,E16.8/
     6          5X,18HCRACK LENGTH      ,E16.8/
     7          5X,18HKMAX APPLIED      ,E16.8/
     8         5X,18HKMAX EFFECTIVE    ,E16.8/
     9         5X,18HDELTA K           ,E16.8/
     /         5X,18HDA/DN             ,E16.8)
      RETURN
      END
      SUBROUTINE K(SIGMA,A,KM)
      COMMON/CORFAC/DUMMY(14),BETA(10),DUMMY1(422)
      COMMON/DATA/ EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AMAX,NZERO
      INTEGER EQN
      REAL NZERO
      REAL KM
      DATA PI/3.14159265/
      CALL BETAS (SIGMA,A,BETAT,Q)
      IF(NASA.NE.0) GO TO 100
      KM=SIGMA*SQRT(PI*A)*BETAT
      RETURN
  100 KM=SIGMA*SQRT(A)*BETAT
      RETURN
      END
      SUBROUTINE BETAS(SIGMA,A,BETAT,Q)
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /                KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /                AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /                BTABL2(100),ASTART(10),ASTOP(10)
      COMMON/MKCRVE/ MK(100)
      REAL MK,MSUBK,M1
      BETAT=1.0
      MSUBK=1.0
      Q=1.0
      PI = 3.14159265
    5 DO 100 I=1,10
      J=IBETA(I)
      IF(J.EQ.0) GO TO 100
      IF(A.LT.ASTART(J). OR .A.GT.ASTOP(J)) GO TO 100
      GO TO(10,20,30,40,50,60,70,80,90),J
C
C     CONSTANT MULTIPLIER
C
   10 BETAT = BETAT * BETA(J)
      GO TO 100
C
C     FINITE WIDTH SECANT CORRECTION
C
   20 HOLE = BETA(5)+BETA(6)
      SMALLC=A
      IF(ISURF.NE.0) SMALLC=A/(2.*RATIO)
      SMALLK = (SMALLC + HOLE)/BETA(J)
      IF(SMALLK.GT.0.5) GO TO 200
      BETAT = BETAT*SQRT(1./COS(PI*SMALLK))
      GO TO 100
C
C     TABULAR CORRECTION FACTOR
C
   30 SMALLK=A/BETA(J)
      BETAT=BETAT*TBLKUP(AOVERB,BTABLE,NPTS,100,SMALLK)
      GO TO 100
C
C     SECOND TABULAR CORRECTION FACTOR
C
   40 SMALLK=A/BETA(J)
      BETAT=BETAT*TBLKUP(AOVRB2,BTABL2,NPTS2,100,SMALLK)
      GO TO 100
C
C     BOWIE SOLUTION FOR SINGLE CRACK FROM CIRCULAR HOLE
C
   50 BETAT = BETAT * (0.6762062 + (0.8733015/(0.3245442+A/BETA(J))))
      GO TO 100
C
C     BOWIE SOLUTION FOR DOUBLE CRACK FROM CIRCULAR HOLE
C
   60 BETAT = BETAT * (0.9438510 + (0.6805078/(0.2771965+A/BETA(J))))
      GO TO 100
C
C     SOLUTION FOR ASTM COMPACT TENSION SPECIMEN USING J. C. NEWMAN
C     EQUATION (12) FROM 'STRESS ANALYSIS OF THE COMPACT SPECIMEN
C      INCLUDING THE EFFECTS OF PIN LOADING'
   70      AW = A/BETA(I)
           BETAT = BETAT*(4.55-40.32*AW+414.7*AW**2.-1698.*AW**3.
     1     +3781.*AW**4.-4287.*AW**5.+2017.*AW**6.)/SQRT(BETA(I)*A*PI)/
     2      THICK
         GO TO 100
C        SOLUTION FOR GRUMMAN COMPACT TENSION SPECIMEN
C      H/W = .95          D/W = .25
   80      AW = A/BETA(I)
           POLY = .1229+16.4098*AW-37.395*AW**2.+54.7667*AW**3.
         IF ( AW .LE. 0.5 ) GO TO 81
           POLY = 114.054-830.132*AW+2327.177*AW**2.-2890.811*AW**3.
     1      +1382.306*AW**4.
   81      BETAT = BETAT*POLY/SQRT(PI*A)/THICK
         GO TO 100
C       SOLUTION FOR COMPACT TENSION SPECIMEN FROM ASTM E 747-83
C       CONSTANT-LOAD-AMPLITUDE FATIGUE CRACK GROWTH RATES ABOVE
C       10-08 M/CYCLE
  90       AW=A/BETA(I)
           BETAT=BETAT*(2.+AW)*(.886+4.64*AW-13.32*AW**2.+14.72*AW**3.
     1     -5.6*AW**4.)/((THICK*(1-AW)**1.5)*SQRT(BETA(I)*PI*A))
  100 CONTINUE
      IF(ISURF.EQ.0) RETURN
C
C     SURFACE FLAW CORRECTION
C     FROM NEWMAN - NASA TN D-8244
```

```
C
      Q=PHI**2.0-0.212*(SIGMA/SIGMAY)**2.0
      AOVERT=A/THICK
      AOVERC=2.*RATIO
      P=2.+8.*AOVERC**3.
      M1=1.13-0.1*AOVERC
      MSUBK=(SQRT(Q/AOVERC)-M1)*AOVERT**P
      BETAT=BETAT*(M1+MSUBK)/SQRT(Q)
      RETURN
  200 WRITE(6,1000)
 1000 FORMAT(1H0, 70(1H*)/ 5X,52HCRACK LENGTH EXCEEDS PLATE WIDTH. TERMI
     1NATE PROBLEM.  / 70(1H*))
      ISTOP = 1
      RETURN
      END
      SUBROUTINE YLDZNE(CYC,A,DN,FR)
C
C     THIS ROUTINE CONTROLS APPLICATION OF THE WHEELER MODEL AND THE
C     EFFECTIVE STRESS(WILLENBORG)MODEL BASED ON THE PROGRESS THRU A
C     YIELD ZONE(ASUBP)DUE TO AN OVERLOAD.
C
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /              NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /               KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      REAL KMAX,KOVLD
      EXTERNAL FR,RATE,DNDA
      CYCF = CYC + DN
      IF(RETARD.EQ.1) GO TO 100
C
C     IS THIS LAYER SUBJECT TO RETARDATION &
C
      IF(SMAX(J4,ISEG).GE.OVLD) GO TO 800
C
C     RETARDATION IS APPLIED.
C
      RETARD = 1
C
C     DETERMINE EXTENT OF ELASTIC-PLASTIC INTERFACE
C
      CALL K(OVLD,A,KOVLD)
      RSUBY = RY(KOVLD,PLSTRN)
      ASUBP = A + RSUBY
C
C     WILL FIRST CYCLE OF THIS LAYER CAUSE ASUBP TO BE EXCEEDED &
C
  100 CALL SIGEFF(A,ASUBP,SIGMAY)
      IF(RETARD.EQ.0) GO TO 800
      CALL K(SIGMAX,A,KMAX)
      RSUBY1 = RY(KMAX,PLSTRN)
      IF(A+RSUBY1.LT.ASUBP) GO TO 200
      RETARD = 0
      GO TO 800
C
C     THIS LAYER IS SUBJECT TO RETARDATION.
C     ASSUME THAT RETARDATION APPLIES OVER THE ENTIRE LAYER.
C
  200 CYC1 = CYC
      AA = A
      CALL RK1DES(CYC1,AA,DN,FR)
      IF(ISTOP.NE.0) RETURN
      IF(AA.GT.ASUBP) GO TO 400
      CALL K(SIGMAX,AA,KMAX)
      RSUBY = RY(KMAX,PLSTRN)
C
C     CHECK  ASSUMPTION.
C
      IF(AA+RSUBY.GE.ASUBP) GO TO 400
      CYC = CYC1
      A = AA
      RETURN
C
C     ENTIRE LAYER IS NOT RETARDED
C     CALCULATE DELTA A AND ITS ASSOCIATED YIELD ZONE(RSUBY) SUCH THAT
C     (A+DA+RSUBY = ASUBP) TO A GIVEN TOLERANCE(TOL)
C
  400 TOL = 1.0E-5
      RED = 1.0
      SUBT = 0.1
      CYC1 = CYC
      AA = A
  500 DA = (ASUBP-(AA+RSUBY1)) * RED
      CALL SIGEFF(AA+DA,ASUBP,SIGMAY)
      CALL K(SIGMAX,AA+DA,KMAX)
      RSUBY = RY(KMAX,PLSTRN)
      IF(AA+DA+RSUBY.LT.ASUBP) GO TO 700
      RED = RED - SUBT
      GO TO 500
  600 RED = RED + SUBT
      SUBT = SUBT/10.
      GO TO 500
  700 IF(SUBT.GT.TOL) GO TO 600
C
C     KNOWING DN, DETERMINE NUMBER OF CYCLES IN THIS LAYER REQUIRED
C     TO PRODUCE DA.
C
      CALL RUNKUT(AA,CYC1,DA,DNDA)
      IF(ISTOP.NE.0) RETURN
      CYC = CYC1
      A = AA
```

```
C
C     REMAINDER OF LAYER IS NOT SUBJECT TO RETARDATION.
C     USE UNRETARDED DADN FUNCTION(RATE)
C
      RETARD = 0
      DN = CYCF - CYC
      IF(DN.LE.0.0) RETURN
C
C     UNRETARDED DADN FUNCTION
C
  800 SIGMAX = SMAX(J4,ISEG)
      SIGMIN = SMIN(J4,ISEG)
      OVLD = SIGMAX
      CALL RK1DES(CYC,A,DN,RATE)
      RETURN
      END
      REAL FUNCTION RY(K,PLSTRN)
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /               KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      REAL K
      INTEGER PLSTRN
      DATA PI,ROOT2 /3.1415926,2.828428/
      IF(K.LE.0.) GO TO 999
      RY = ((K/SIGMAY)**2.)/(2.*PI)
      IF(PLSTRN.NE.0) RY = RY/ROOT2
      RETURN
  999 RY= 0.0
      RETURN
      END
      SUBROUTINE DNDA(A,CYC,DDNINV)
C
C     THIS ROUTINE CALCULATES THE NUMBER OF CYCLES IN A LAYER OVER
C     WHICH RETARDATION IS APPLIED
C
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      GO TO (10,20),MODEL
   10 CALL WHELER(CYC,A,DADN)
      GO TO 30
   20 CALL WLNBRG(CYC,A,DADN)
   30 DDNINV = 1.0/DADN
      RETURN
      END
      SUBROUTINE WLNBRG(CYC,A,DADN)
C
C     THIS ROUTINE BRINGS THE EFFECTIVE STRESSES GENERATED BY THE
C     WILLENBORG MODEL INTO THE UNRETARDED GROWTH RATE EQUATIONS.
C
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/MDATA/ MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     /                KSUBQ,SIGMAY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ
      CALL SIGEFF(A,ASUBP,SIGMAY)
      CALL RATE(CYC,A,DADN)
      RETURN
      END
      SUBROUTINE SIGEFF(A,ASUBP,SIGMAY)
C
C     THIS ROUTINE COMPUTES THE EFFECTIVE STRESSES FOR USE IN THE
C     WILLENBORG MODEL. THE EFFECTIVE STRESSES ARE STORED IN LOCATIONS
C     'SIGMAX' AND 'SIGMIN' IN COMMON BLOCK /RDATA/.
C
C
      COMMON/DATA/IDUM1,NASA,IDUM2(5),DUM3(3)
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,DUMMY,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/LDATA/SMAX( 20,10),SMIN( 20,10),CYCLES( 20,10),NLYRS(10),
     /               NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      COMMON/MDATA/MATID(18),C,SMALLN,CARRAY(100),SNARAY(100),KSUBC,
     1              KSUBQ,DUMMYY,DELKTH,RMULT,RCUT,OLMAX
      REAL KSUBC,KSUBQ,KMAX
C
C     PUT APPLIED STRESSES IN SIGMAX AND SIGMIN
C
      SIGMAX = SMAX(J4,ISEG)
      SIGMIN = SMIN(J4,ISEG)
      IF(MODEL.EQ.2) GO TO 100
      RETURN
  100 CALL BETAS(SIGMAX,A,BETAT,QMAX)
      IF(A.GT.ASUBP) GO TO 200
      SIGREF =  (SIGMAY*SQRT(2.0*(ASUBP-A)/A))/BETAT
      IF(NASA.NE.0) SIGREF=SIGREF*SQRT(3.1415926)
      IF(PLSTRN.NE.0) SIGREF = SIGREF * SQRT(2.828428)
      SIGRED = SIGREF - SIGMAX
      CALL DELTA(A,DELTAK,KMAX,R)
      THRSLD = DELKTH*(1.-RMULT*R)
      PHI = (1.-THRSLD/KMAX)/(OLMAX-1.)
      IF(OLMAX.EQ.0.) PHI = 1.
      SIGRED = PHI * SIGRED
      IF(SIGRED.LE.0.0) GO TO 200
      SIGMAX = SIGMAX - SIGRED
      IF(SIGMAX.LT.0.0) SIGMAX = 0.0
      SIGMIN = SIGMIN - SIGRED
      IF(SIGMIN.LT.0.0) SIGMIN = 0.0
      RETURN
  200 RETARD = 0
      RETURN
      END
      SUBROUTINE WHELER(CYC,A,DADN)
C
C     THIS ROUTINE APPLIES THE WHEELER CORRECTION TO THE UNRETARDED
```

```
C     GROWTH RATE
C
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,ASUBP,SMALLM
      INTEGER RETARD,PLSTRN
      CSUBP = 1.0
C
C     DETERMINE EXTENT OF CURRENT YIELD ZONE
C
      CALL K(SIGMAX,A,KMAX)
      RSUBY = RY(KMAX,PLSTRN)
      IF(A + RSUBY .GE. ASUBP) GO TO 20
C
C     CALCULATE WHEELER'S RETARDATION PARAMETER
C
      CSUBP = (RSUBY/(ASUBP-A))** SMALLM
   20 CALL RATE(CYC,A,DADN)
      DADN = CSUBP * DADN
      RETURN
      END
      SUBROUTINE RK1DES(CYC,A,DCYC,F)
C
      EXTERNAL F
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORTRD
      IF(DCYC .GE. 1.) GO TO 300
      CYCF=CYC+DCYC
      A0=A
   50 H=0.005*A0
      A1=A0+H
      AGROW=A0+2.0*H
      CALL F(CYC,A1,DADN)
      IF(DADN.LE.0.) GO TO 200
      IF(ISTOP.NE.0) GO TO 250
      DCYCR=2.*H/DADN
      IF(DCYCR.GT.DCYC) GO TO 100
      CYC=CYC+DCYCR
      DCYC=DCYC-DCYCR
      A0=AGROW
      GO TO 50
  100 DA=(CYCF-CYC)*DADN
      A0=A0+DA
  200 CYC=CYCF
      A=A0
  250 RETURN
  300 CALL RUNKUT(CYC,A,DCYC,F)
      RETURN
      END
      FUNCTION TBLKUP(X,Y,N,NMAX,ARG)
C
C
       DIMENSION X(NMAX),Y(NMAX)
      DO 10 I=1,N
      IF(X(I)-ARG) 10,20,20
   10 CONTINUE
      I=N
   20 IF(I-1)30,30,40
   30 I=2
   40 SLOPE=(Y(I)-Y(I-1))/(X(I)-X(I-1))
      TBLKUP=SLOPE*(ARG-X(I-1))+Y(I-1)
      RETURN
      END
      SUBROUTINE CLOSUR(CYC,A1,DN)
C
C         EQUIVALENCE BETWEEN CRACKS 2  AND  CLOSUR
C        CRACKS 2              CLOSUR
C        ASUBP                 AP
C        SMAX                  S
C        CYCLES                CYCLS
C        ISEG                  I
C        J4                    K
C        SMALLN                AN    (NOT USED AS SMALLN ANYWAY)
C        KSUBQ                 AKC
C        DELKTH                KTH
C        A                     A1
C        AMAX                  AF
C
C
      COMMON/DATA/ EQN,NASA,J1PR,J2PR,J3PR,J4PR,J5PR,AZERO,AF   ,NZERO
      INTEGER EQN
      REAL NZERO
      COMMON/RDATA/ MODEL,RETARD,PLSTRN,OVLD,SIGMAX,SIGMIN,AP,SMALLM
      INTEGER RETARD,PLSTRN
      COMMON/MDATA/ MATID(18),C,AN,CARRAY(100),SNARAY(100),KSUBC,
     /                AKC,SIGMAY,    KTH,RMULT,RCUT,OLMAX
      REAL KSUBC
      COMMON/LDATA/   S( 20,10),SMIN( 20,10),CYCLS( 20,10),NLYRS(10),
     /              NBLKS,IBLKS( 50 ),ISEGS( 50 ),NSEGS
      COMMON/STEPS/   K,J1,J2,J3,I,J5,ISTOP,NORTRD
      COMMON/CORFAC/ ISURF,RATIO,PHI,THICK,IBETA(10),BETA(10),NPTS,
     /               AOVERB(100),BTABLE(100),NPTS2,AOVRB2(100),
     /               BTABL2(100),ASTART(10),ASTOP(10)
      COMMON/PARIS/C1,SN1,DKCOM,C2,SN2
      COMMON /DIRECT/ NDADN
      COMMON/OUTPOT/XK,XKA,XKEFFN,IFLT,DADNPR
      COMMON/CLOS/CF,CCOEF,CFEXP,B,BOL,NSAT
      COMMON/CLOSIC/SC,SPEAK,PRVMX,APEAK,SC1,SC2,SC3,PRVMN
      REAL      ISUM,KTHSG,KCOEF,KEXP,NOL,NSAT,NPR,NPREV,KTH
      DIMENSION Q(6),QQ(2)
      DATA PI/3.14159265/
C
C
C
C             CLOSURE FUNCTION
          CLOSE(G1,G2) = G1*(CCOEF + (CF - CCOEF)*(1. + G2)**CFEXP)
```

```
C
C             FUNCTION FOR SC  LT  OR  EQ  SC INITIAL
          DOWN(Z1,Z2,Z3) = Z1 - (Z1-SC3)*(Z2/Z3)**B
C
C             FUNCTIONS FOR SC  GT  SC INITIAL
          SCONE(SC3) = SC3*BOL
C
C             FUNCTION FOR INCREASING CLOSURE STRESS
          NPREV(SC,SC3,SC11) =1.+(NSAT-1.)*(SC-SC11)/(SC3-SC11)
C
C
C
C INITIALIZE PARAMETERS
          IGROW = 0
          MODE = 0
          ITEM = 1
          IGROW = 1
          CYSUM = CYCLS(I,K)
          NOL = 0.
          ASTRT = A1
          KLU = 1
          R = 0.
        IF ( S(I,K) .NE. 0. )  R = SMIN(I,K)/S(I,K)
          SMNGR = SMIN(I,K)
        IF ( SPEAK .NE. 0. ) GO TO 30
      SINITL = S(I,K)
      IF ( SINITL .LE. 0.) SINITL = 0.05*SIGMAY
      R = SMIN(I,K)/SINITL
      CALL BETAS(S(I,K),A1,ALF,QE)
          XK = SINITL*SQRT(PI*A1)*ALP
          G1 = SINITL
          G2 = R
      IF ( G2 .LT. -1.) G2 =-1.
          SC = CLOSE(G1,G2)
          SC1 = SC
          SC2 = SC
          SC3 = SC
          SPEAK = SINITL
          PRVMX = SINITL
          PRVMN = SMIN(I,K)
          APEAK = A1
          AP = A1 + RY(XK,PLSTRN)
          OMGA2 = AP
   30 CONTINUE
C
C
C START ANALYSIS
  100     ISUM =0.
C
C
          G1 = S(I,K)
          G2 = R
      IF ( G2 + 1. ) 104,105,105
 104    G2 = -1.
 105    SC3 = CLOSE(G1,G2)
      IF(SMIN(I,K) - PRVMN) 5003,60,60
C            MINIMUM STRESS ADJUSTMENT
 5003 CONTINUE
        PRVMN = SMIN(I,K)
        G1 = SPEAK
        G2 = SMIN(I,K)/SPEAK
      IF (G2 + 1.) 5005,5006,5006
 5005   G2 = -1.
 5006   SC1 = CLOSE(G1,G2)
        G1 = PRVMX
        G2 = SMIN(I,K)/PRVMX
      IF (G2 + 1.) 57,58,58
   57   G2 = -1.
   58   SC3 = CLOSE(G1,G2)
        G3 =  ASTRT - APEAK
        G4 = AP - APEAK
        IF(G3/G4 .LT. 0.) G3=0.
        SC2T = DOWN(SC1,G3,G4)
      IF(SC2T - SC2) 59,60,60
   59   SC2 = SC2T
   60   SC = SC2
      IF(S(I,K)-SC2) 450,450,5009
 5009   SMNGR = SC2
      IF (SMIN(I,K)-SC2) 5010,5011,5011
 5010   SMNGR = SC2
 5011 CALL BETAS(S(I,K),A1,ALP,QE)
        XKEFF=(S(I,K)-SMNGR)*SQRT(PI*A1)*ALP
        CKTH = KTH*(1.-RMULT*R)
        IF (R .LT. 0. ) CKTH = KTH
     IF(XKEFF*(1.-R)/(1.-CF) .LE. CKTH) GO TO 450
 5012   G1 = S(I,K)
        G2 = R
      IF (G2 + 1.) 93,94,94
   93   G2 = -1.
   94   SC3 = CLOSE(G1,G2)
      IF ( S(I,K) - SC3 ) 95,96,96
   95   SC3 = S(I,K)
   96 CONTINUE
      IF(S(I,K) - SPEAK) 90,5013,5013
 5013   KLU = 3
C
C          INITIALIZATION FOR INTEGRATION ROUTINE
C
C     MODE = 1   IF  CYCLS(I,K) .LT. 20 AND SC3 .LE. SC
C     MODE = 2   IF  CYCLS(I,K) .LT. 20 AND SC3 .GT. SC
C     MODE = 3   IF  CYCLS(I,K) .GE. 20, SC3.GT.SC AND N.LT.NSAT
C     MODE = 4   IF  CYCLS(I,K) .GE. 20 AND SC3 .LT. SC
C     MODE = 5   IF  CYCLS(I,K) .GF. 20 AND SC3 = SC
C     INTEGRATION PERFORMED FOR MODE = 4 AND 5
```

```
   90     MODE = 5
        IF ( CYCLS(I,K) - 20. ) 201,203,203
  201     MODE = 1
        IF ( SC - SC3 ) 202,205,205
  202     MODE = 2
        GO TO 205
  203   IF ( SC .EQ. SC3 ) GO TO 205
          MODE = 3
        IF ( SC3 - SC ) 204,205,205
  204     MODE = 4
  205 CONTINUE
C
C
          DCYC = CYCLS(I,K)/30000.
        IF ( DCYC .LT. 1.0 )  DCYC = 1.0
          NNCYC = CYCLS(I,K)/DCYC
C
C       START CYCLE - BY - CYCLE ANALYSIS
C
      DO 310 J = 1,NNCYC
        IF ( MODE - 3 ) 210,210,6000
  210   IF (SC3 - SC) 92,92,111
   92   GO TO (101,160,101,101),KLU
  101   IF (OMGA1) 5014,5015,5015
 5014     OMGA1 = 0.
 5015   IF (OMGA1/OMGA2 - 1.) 5017,5017,5016
 5016     OMGA1 = OMGA2
 5017     IF (OMGA1/OMGA2 .LT. 0.) OMGA1 =0.
          SC = DOWN(SC2,OMGA1,OMGA2)
  111   IF(SMIN(I,K) - SC) 5023,160,160
 5023     SMNGR = SC
  160 CONTINUE
        GO TO 4050
 4051 CONTINUE
          ISUM = ISUM + DCYC
          CYC = CYC + DCYC
          CYSUM = CYSUM - DCYC
          DN = CYSUM
        GO TO (138,501,501,501),ITEM
  138 CONTINUE
        IF ( MODE - 2 ) 143,144,144
  143     OMGA1 = A1 - ASTRT
        IF (A1 - AP) 300,141,141
  141     KLU = 2
          SC = SC3
          AP = A1
        GO TO 300
  144     SC11 = SCONE(SC3)
          NPR = NPREV(SC,SC3,SC11)
        IF ( NPR ) 145,146,146
  145     NPR = 0.
  146     NOL = NPR + DCYC
```

```
          SC = SC11 + (SC3-SC11)*(NOL-1.)/(NSAT-1.)
        IF ( SC3 - SC ) 147,147,300
  147     SC = SC3
        IF ( MODE - 2 ) 300,300,148
  148     MODE = 5
  300 CONTINUE
  310 CONTINUE
  311 CONTINUE
          PRVMX = S(I,K)
          SC2 = SC
        GO TO (501,400,400,400),KLU
  400 CALL BETAS(S(I,K),A1,ALP,QE)
          XK = S(I,K)*SQRT(PI*A1)*ALP
          AP = A1 + RY(XK,PLSTRN)
          APEAK = A1
          SPEAK = S(I,K)
          PRVMN = SMIN(I,K)
        IF (SC3 - SC) 420,420,430
  420     SC1 = SC3
        GO TO 501
  430     SC1 = SC
        GO TO 501
  450     ISUM = ISUM + CYCLS(I,K)
          CYC = CYC + CYCLS(I,K)
          DN = 0.
  501 CONTINUE
C
        GO TO(500,600,620,503),ITEM
  503     ISTOP = 2
          A1 = ATRANS/(2.*RATIO)
      WRITE(6,1) A1,CYC
    1 FORMAT(1H0, 70(1H*)/ 5X,55HTRANSITION TO A THRU CRACK OF EFFECTIVE
     1 LENGTH, AEFF =  ,F9.5,4H AT ,F12.2,7H CYCLES  /,1X, 70(1H*))
        GO TO 50
C
C
C      END OF CYCLE BY CYCLE ROUTINE
C
  500 CONTINUE
        GO TO 50
C          A1 EXCEEDED AF  (AMAX)
  600     ISTOP = 1
        GO TO 50
  620     ISTOP = 1
      WRITE(6,7)
    7 FORMAT(1H0, 70(1H*)/ 5X,46HKMAX APPLIED EXCEEDS KSUBQ. PROBLEM TER
     1MINATED/1X, 70(1H*)/1H0, 1X,26HLAST CALCULATED VALUES ARE///)
          DELTAK = XK*(1.-R)
      WRITE(6,8)   J1,J2,I,IFLT,K,CYC,A1,XK,XKEFF,DELTAK,DADN
    8 FORMAT( 5X,18HBLOCK IN SPECTRUM ,I4/
     /        5X,18HSEGMENT NUMBER    ,I4/
     /        5X,18HMISSION NUMBER    ,I4/
     /       5X,18HFLIGHT NUMBER     ,I4/
     /       5X,18HLAYER IN MISSION  ,I4/
     5       5X,18HACCUMULATED CYCLES,E16.8/
     6       5X,18HCRACK LENGTH      ,E16.8/
     7       5X,18HKMAX APPLIED      ,E16.8/
     8       5X,18HKMAX EFFECTIVE    ,E16.8/
     9       5X,18HDELTA K           ,E16.8/
     /       5X,18HDA/DN             ,E16.8)
C
C
   50 CONTINUE
          XKA = XK
      RETURN
C
C
C          GROWTH CALCULATIONS
 4050 CONTINUE
C          CHECK FOR XK (KMAXA) .GE. AKC (KSUBQ)
      CALL BETAS(S(I,K),A1,ALP,QE)
          XK = S(I,K) *SQRT(PI*A1)*ALP
        IF(XK - AKC) 4062,4060,4060
 4060     ITEM = 3
          DADN = 0.
        GO TO (502,502,502,4062,4062),MODE
 4062     XKEFF = (S(I,K) - SMNGR)*SQRT(PI*A1)*ALP
          XKEFFN = XKEFF/(1.-CF)
        IF ( EQN .EQ. 1 ) GO TO 9000
        IF ( EQN .EQ. 4 ) GO TO 9010
        IF ( EQN - 2 ) 8000,8000,8010
 8000 CONTINUE
C          BI-LINEAR PARIS EQUATION
          C = C1
          AN = SN1
        IF ( XKEFFN .GE. DKCOM ) C = C2
        IF ( XKEFFN .GE. DKCOM ) AN = SN2
          DADN = C*XKEFFN**AN
        GO TO 8020
C
 8010 CONTINUE
C          TABULAR RATE VALUES
          DADN = TBLKUP(CARRAY,SNARAY,NDADN,100,XKEFFN)
          DADN = 10.**DADN
 8020 CONTINUE
      IF(RETARD.NE.0)DADNPR = DADN
          A1 = A1 + DADN*DCYC
        IF (A1 - AF)4064,4063,4063
 4063     ITEM = 2
        GO TO 502
 4064   IF ( ISURF .EQ. 0 )  GO TO 502
          ATRANS = THICK - (((XK/SIGMAY)**2.)/(2.*PI))
        IF ( A1 .LT. ATRANS) GO TO 502
          ITEM = 4
```

```
  502  GO TO (4051,6060),IGROW
C
C
C          END OF CRACK GROWTH CALCULATIONS
C
C
C
C         INTEGRATION ROUTINE
 6000 CONTINUE
       IF ( MODE - 4 ) 6010,6010,6020
 6010    Q(1) = A1
         Q(2) = AP
         Z2 = A1 - ASTRT
         Z3 = AP - ASTRT
         IF ( Z2/Z3 .LT. 0.) Z2 =0.
         QQ(1) = DOWN(SC2,Z2,Z3)
         QQ(2) = SC3
       IF ( ISURF .EQ. 0 )  GO TO 6030
       IF ( Q(2) .LT. THICK )  GO TO 6030
         Q(2) = THICK
         Z2 = THICK - ASTRT
         IF (Z2/Z3 .LT. 0.) Z2 =0.
         QQ(2) = DOWN(SC2,Z2,Z3)
         ISTOP = 2
       GO TO 6030
 6020    Q(1) = A1
         Q(2) = 1.10*A1
         QQ(1) = SC3
         QQ(2) = SC3
       IF ( ISURF .EQ. 0 )  GO TO 6030
       IF ( Q(2) .LT. THICK )  GO TO 6030
         Q(2) = THICK
         ISTOP = 2
 6030 CONTINUE
         SC = QQ(1)
       DO 6100 KK = 1,2
         A1 = Q(KK)
         SMNGR = SMIN(I,K)
       IF ( SMIN(I,K) - QQ(KK) ) 6045,6050,6050
 6045    SMNGR = QQ(KK)
 6050    IGROW = 2
       GO TO 4050
 6060  GO TO (6070,6065,6065,6065),ITEM
 6065  GO TO (501,6068),KK
 6068    ITEM = 1
         ISTOP = 0
 6070    KKK = KK + 2
         KKKK = KK + 4
         Q(KKK) = XKEFF
         Q(KKKK) = DADN
 6100 CONTINUE
         Q2 = (Q(3)-Q(4))/(Q(1)-Q(2))
```

```
          Q1 = Q(3) - Q2*Q(1)
          Q4 = (ALOG(Q(5)/Q(6)))/(ALOG(Q(3)/Q(4)))
          Q3 = Q(5)/(Q(3)**Q4)
          Q5 = 1. - Q4
          DELTN = (Q(4)**Q5 - Q(3)**Q5)/(Q2*Q3*Q5)
        IF ( DELTN - CYSUM ) 6200,6150,6250
 6150 CONTINUE
 6200     A1 = Q(2)
        IF ( AP - A1 ) 6210,6210,6220
 6210     AP = A1
          KLU = 2
          MODE = 5
 6220     ISUM = ISUM + DELTN
          CYC = CYC + DELTN
          CYSUM = CYSUM - DELTN
          DN = CYSUM
        IF ( CYSUM - .01 ) 6260,6240,6240
 6240   IF ( MODE - 4 ) 6010,6010,6020
 6250     Q(4) = (Q(3)**Q5+CYSUM*Q2*Q3*Q5)**(1./Q5)
          Q(2) = (Q(4) -Q1)/Q2
          DELTN = CYSUM
          ISTOP = 0
        GO TO 6150
 6260     SC = SC3
 6270     Z2 = A1 - ASTRT
          Z3 = AP - ASTRT
        IF (Z3 .LE. 0. ) Z3 = Z2
          IF (Z2/Z3 .LT. 0.) Z2 =0.
          SC = DOWN(SC2,Z2,Z3)
 6280   GO TO 311
C
C
C         END OF INTEGRATION ROUTINE
C
C
C       DIAGNOSTICS FOR IMPROPER EQN
 9000 WRITE(6,9005)
 9005 FORMAT(1H0, 1X,44HCLOSURE MODEL CAN NOT ACCEPT FORMAN EQUATION /
     1    40X,20HEXECUTION SUPPRESSED )
        GO TO 9020
 9010 WRITE(6,9015)
 9015 FORMAT(1H0, 1X,44HCLOSURE MODEL CAN NOT ACCEPT WALKER EQUATION /
     1    40X,20HEXECUTION SUPPRESSED )
 9020     ISTOP = 1
          XKA = XK
      RETURN
C
C
      END
      SUBROUTINE RUNKUT(X,Y,DX,F)
C
      EXTERNAL F
      COMMON/STEPS/ISEG,J1,J2,J3,J4,J5,ISTOP,NORETRD
   10 X0=X
      X=X+DX
      H=DX
   20 IF(ABS(H).GT.ABS(X-X0)) H=X-X0
   30 Y0=Y
      HT=H
      XT=X0
      RMAXP=1.E37
   40 YT=Y0
      ASSIGN 50 TO K
      GO TO 100
   50 CONTINUE
   60 YP=Y
   70 HT=0.5*H
      ASSIGN 80 TO K
      GO TO 100
   80 CONTINUE
   90 YT=Y
      XT=X0+HT
      ASSIGN 150 TO K
  100 CALL F(XT,YT,P0)
      IF(ISTOP.EQ.0) GO TO 110
      X = XT
      Y = YT
      RETURN
  110 Y=YT+0.5*HT*P0
      CALL F(XT+0.5*HT,Y,P1)
      IF(ISTOP.EQ.0) GO TO 120
      X = XT+0.5*HT
      RETURN
  120 Y=YT+0.5*HT*P1
      CALL F(XT+0.5*HT,Y,P2)
      IF(ISTOP.EQ.0) GO TO 130
      X = XT+0.5*HT
      RETURN
  130 Y=YT+HT*P2
      CALL F(XT+HT,Y,P3)
      IF(ISTOP.EQ.0) GO TO 140
      X = XT+HT
      RETURN
  140 Y=YT+HT*(P0+2.*(P1+P2)+P3)/6.
      GO TO K,(50,80,150)
  150 RMAX=0.
  160 RMAX=AMAX1(RMAX,0.07*ABS((Y-YP)/Y))
      IF((RMAX.GT.1.E-07).AND.(RMAX.LT.RMAXP)) GO TO 170
      X0=X0+H
      IF(X0.EQ.X) RETURN
      IF((RMAX.LT.1.E-08). OR.(RMAX.GT.RMAXP)) H=H+H
      GO TO 20
  170 H=HT
      XT=X0
```

```
180 YP=YT
190 YT=YO
    RMAXP=RMAX
    GO TO 70
    END
```

Sample_Input

```
TITLE
    1
LOW-MED-HIGH SPECIMEN 84-502, EE3, WHEELER M=6.0, 1 CYCLE=20 SEC
EQUATION
SIGMOID
MATERIAL
INCO 718 MSE FIT DA/DT TIMES 20
21.0      272.73    -5.6942677-1.1      1.8       -1.8
300.0     120.0
THRESHOLD
21.0      1.0
LIMITS
.4087     1.1330    0.        0.0
ANALYSIS
RETARD    1.0       0.0       6.0       1.0
BETA      9.0       1.5736    .394
END
LOADS
    1    0 PROOF TEST SPECTRUM
MAX-MIN    SUSTAINED LOAD EQUIVALENT OF 1 CYCLE = 20 SEC R=.4
          2.740     1.096     1.
END
MAX-MIN    OVERLOAD OF 20 PERCENT EQUIVALENT CYCLE
          3.288     0.0       1.
END
END LOADS
SPECTRUM
    7    0
   72    1
    1    2
 1573    1
    1    2
  590    1
    1    2
 1000    1
RESTART
    1
PRINT
    1    0    0    1
END DATA
```

Vita

Robert L. Hastie Jr. was born on 18 November 1957 in Harrison, Pennsylvania. He graduated fron Kiski Area high school in Vandergrift, Pennsylvania in 1975 and attended Grove City College. In, May 1979 he graduated Cum Laude and received a Bachelor of Science degree in Mechanical Engineering. Upon graduation, he received a commission in the USAF through the ROTC program. He was called to active duty in August 1979 and served as a Structural Strength Engineer in the Deputy for Engineering at the Aeronautical Systems Division (ASD) at Wright-Patterson AFB, Ohio. In November 1980 he was transfered to the Deputy for Propulsion within the ASD and was a Propulsion Durability Engineer for the F107, F100, TF-34, and F100 engines. In June 1984, Captain Hastie entered the School of Engineering, Air Force Institute of Technology, to earn a Master of Science degree in Astronautical Engineering.

Permanent address: RD 1 Box 295
Vandergrift, Pa 15690

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A164018

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
AFIT/GA/AA/85D-6

5. MONITORING ORGANIZATION REPORT NUMBER(S)

6a. NAME OF PERFORMING ORGANIZATION
School of Engineering

6b. OFFICE SYMBOL *(If applicable)*
AFIT/ENY

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State and ZIP Code)*
Air Force Institute of Technology

7b. ADDRESS *(City, State and ZIP Code)*

8a. NAME OF FUNDING/SPONSORING ORGANIZATION

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State and ZIP Code)*

10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
61102F	2302	P1	01

11. TITLE *(Include Security Classification)*
See Box 19

12. PERSONAL AUTHOR(S)
Robert L. Hastie Jr., B.S., Captain, USAF

13a. TYPE OF REPORT
MS THESIS

13b. TIME COVERED
FROM ________ TO ________

14. DATE OF REPORT *(Yr., Mo., Day)*
1985 December

15. PAGE COUNT
139

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB. GR.
20	11	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Sustained-load, Crack Growth Rate, Crack Growth Retardation Elevated Temperature, Nickel-base Superalloy, CRACKS, Crack Growth Prediction Model, Wheeler, Willenborg

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

TITLE: PREDICTING THE EFFECTS OF OVERLOADS ON SUSTAINED-LOAD CRACK GROWTH IN A HIGH-TEMPERATURE SUPERALLOY

THESIS CHAIRMAN: George K. Haritos, Major, USAF

ABSTRACT: See Reverse

Approved for public release: IAW AFR 190-1.
Lynn Wolaver 16 JAN 86
LYNN E. WOLAVER
Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. □ DTIC USERS □

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL
George K. Haritos, Associate Professor

22b. TELEPHONE NUMBER *(Include Area Code)*
(513)-255-2998

22c. OFFICE SYMBOL
AFIT/ENY

EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

Abstract

This ~~study~~ thesis investigates methods of modeling the effects of overloads on high-temperature sustained-load crack growth. In addition to a model previously developed for this specific problem, a computer program developed for low-temperature, high-frequency cyclic load applications was evaluated. Sustained-load hold times were converted to equivalent fatigue cycles to analyze a load spectrum, consisting of sustained-load with periodic overloads. The CRACKS crack growth program was used with the Wheeler and Willenborg models used to account for crack growth retardation due to overloads.

Predictions were compared with experimental test data generated on specimens of Inconel 718 at 650 C with periodic overloads of either 20 or 50 percent. Crack measurements were made using a electric potential system. The application of the electric potential system to crack growth measurement following overloads was extensively evaluated. It was concluded that the system had to be recalibrated after each overload due to a sudden advancement in crack length.

The retardation models were found to require empirical parameters that depend upon the stress intensity level for each overload application. Using relationships developed for these parameters, the CRACKS program using the Wheeler model was found to be capable of predicting the time-to-failure for sustained-loading with periodic overloads within 20 percent of test data. The Willenborg model was found to be inapplicable to this problem because it depends solely on stress ratio which has no physical meaning for sustained-loading. The Wheeler model, on the other hand, could generally be applied to sustained-load crack growth using equivalent fatigue cycles. In conjunction with the CRACKS computer program, this could provide a powerful new method for evaluating crack growth under general engine mission spectra including the effects of overloads.

END

FILMED

3-86

DTIC